NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

FINITE ELEMENT ELECTROMAGNETIC
SCATTERING: AN INTERACTIVE
MICRO-COMPUTER ALGORITHM

by

Edward Michael Connolly

June 1988

Thesis Advisor: M. Morgan

Approved for public release; distribution is unlimited

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b. OFFICE SYMBOL (If applicable) 62	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000	7b. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO / WORK UNIT ACCESSION NO

11. TITLE (Include Security Classification) FINITE ELEMENT ELECTROMAGNETIC SCATTERING: AN INTERACTIVE MICRO-COMPUTER ALGORITHM

12. PERSONAL AUTHOR(S)
CONNOLLY, Edward M.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Master's Thesis	FROM ______ TO ______	1988 June	210

16. SUPPLEMENTARY NOTATION The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. government

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Electromagnetic scattering; Finite element algorithm; Azimuthal potentials; Computer aided design

19. ABSTRACT (Continue on reverse if necessary and identify by block number)
The goal of this thesis effort was to develop an interactive electromagnetic scattering algorithm for personal computer use in the analysis and computer aided design (CAD) of scattering structures. Extensive interactive computer aided design type features have been added to a previously developed finite element computer algorithm which considers scattering from inhomogeneous axisymmetric objects. These computer aided design features allow structures composed of multiple dielectric and/or magnetic materials to be graphically constructed or modified by the user by way of either a keyboard or mouse interface. Bistatic scattering patterns are graphically presented upon completion of the computation for a variety of user specified incident field orientations.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL MORGAN, M.	22b. TELEPHONE (Include Area Code) 408-646-2677 22c. OFFICE SYMBOL 62Mo

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

☆ U.S. Government Printing Office: 1986-606-

UNCLASSIFIED

Approved for public release; distribution is unlimited

Finite Element Electromagnetic Scattering: An Interactive Micro-Computer Algorithm

by

Edward Michael Connolly
Lieutenant, United States Navy
B.S. Math, United States Naval Academy, 1980

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL

ABSTRACT

The goal of this thesis effort was to develop an interactive electromagnetic scattering algorithm for personal computer use in the analysis and computer aided design (CAD) of scattering structures. Extensive interactive computer aided design type features have been added to a previously developed finite element computer algorithm which considers scattering from inhomogeneous axisymmetric objects. These computer aided design features allow structures composed of multiple dielectric and/or magnetic materials to be graphically constructed or modified by the user by way of either a keyboard or mouse interface. Bistatic scattering patterns are graphically presented upon completion of the computation for a variety of user specified incident field orientations.

C715 c6

TABLE OF CONTENTS

I. INTRODUCTION

The study of resonance region electromagnetic interaction with arbitrarily shaped and often inhomogeneous penetrable objects has increased dramatically in the past few years. This new concentration of effort is driven by the likelihood of expanded implementation of high frequency over-the-horizon radars, very high frequency satellite-borne radars and medium frequency surface wave radars. Such radars have the potential to observe "stealth" type platforms whose low radar cross section (RCS) design was limited to more conventional radar frequencies, above 500 MHz. In addition, these new studies of electromagnetic interaction have proven to be of benefit in other areas such as propagation through rain, power absorption by biological tissues, focusing by microwave lenses, radiation from reentry vehicles and electromagnetic pulse (EMP) coupling into vehicles, installations and weapon systems. [Refs. 1 and 2]

As frequencies become sufficiently low, structures having a low radar cross section may exhibit strong resonance characteristics. Design of such structures is complicated by these low frequency resonance characteristics, due to the lack of accurate analytical methods. Even attempts at structure design utilizing

"shaping" and "edge treatments" are ineffective at providing a highly reduced radar cross section at resonant frequencies. At higher frequencies, "asymptotic" formulations are available to rectify these problems [Ref. 3]. A more comprehensive approach is needed, one which allows the designer to rigorously evaluate the interaction of the scatterer with the illuminating field. At the same time, such a tool should be highly interactive with the user.

The subject of this thesis is the development of a generalized, interactive, finite element scattering algorithm to be used in the analysis and computer aided design of scattering structures. In this, the first stage of effort, the programs developed for use on a personal computer calculate electromagnetic scattering by axisymmetric inhomogeneous structures. These structures are composed of user specified lossy or lossless penetrable (i.e., dielectric or magnetic) materials. The resulting programs provide significant improvement in computational speed and versatility over the current integral equation based computer codes being used. The physical structure can be quite general, other than the requirement for rotational symmetry. This general construction avoids the computational restrictions found in surface integral equation formulations, where, as frequency and geometric complexity increase, the time required to obtain a solution

increases. The finite element algorithms, based upon a volume variational formulation, are relatively immune to structural complexity driving the computation time. Another advantage of these algorithms is their method of matrix manipulation. The finite element algorithms produce very "sparse" matrix structures, whereas the integral equation formulations produce "full" matrices [Ref. 4]. Finally, when considering the construction of realistic material configurations, the computational time required utilizing the finite element algorithms can be substantially less than the time required by the use of integral equation formulations [Ref. 5].

The development of the axisymmetric finite element program is based upon Morgan's previous effort [Ref. 1], which restricted structures to non-metallic configurations. Major programming modifications to improve computational efficiency and to incorporate computer aided design features such as interactive graphics are now integral parts of the finite element computer program.

A brief overview of the finite element scattering algorithm, which constitutes the basis of this thesis, is presented in Chapter II. In addition to the appropriate background information, the concept of coupled azimuthal potentials, the finite element algorithm and the unimoment method are reviewed.

Having established, in Chapter II, the necessary theory utilized in the construction of the original version of the computer program EMSCAT, Chapter III presents methods to construct program input data. Included in this chapter are discussions of computer programs written specifically for a given geometric shape, data file input and keyboard input. Basic flow charts provide an overview of the input-output interactions of the all computer programs.

The first half of Chapter IV discusses the extension of the methods used to construct program input data to those methods that utilize graphics. Of special interest is the incorporation of computer aided design features. The second half of this chapter addresses the user interactive output graphics features provided. These features allow the user to obtain one of four different graphs, either on the computer screen or printer, of the EMCAD output data of choice.

In Chapter V, the validation of modifications and improvements to EMCAD is presented. Graphical comparisons are made between the output of the modified version of EMSCAT versus both theoretical calculations and actual experimental data. In addition, validation of new programs written specifically to be used in conjunction with EMCAD is presented.

The brief conclusions in Chapter VI will summarize the work previously presented, as well as provide

recommendations for future modifications to the interactive package of computer programs being used.

The Appendices provide copies of the source code for all the computer programs utilized in this interactive package. A sample listing of the output of EMCAD is attached. As an added feature, user manuals with hints, warnings and directions for all of the major computer programs are included.

II. FINITE ELEMENT SCATTERING ALGORITHM

A. BACKGROUND

In recent years, great attention and effort has been directed toward the attempt to obtain the numerical electromagnetic scattering solutions for inhomogeneous material structures. Early efforts dealt primarily with improving the efficiency of existing computational algorithms. These algorithms utilized surface integral equation formulations, which required the inversion of large "full" matrices. Accordingly, methods using these formulations were computationally slow while being highly restricted in the level of inhomogeneity that could be handled. [Refs. 1 and 2]

A major advance in matrix manipulation occurred with the development by Mei [Ref. 6] of the unimoment method. This new method decoupled the interior and exterior regions associated with radiation or scattering problems, resulting in uniformly banded system matrices for the interior region solution. The primary advantage of uniformly banded matrices is the capability to invert them directly using fast banded matrix inversion techniques, such as the Riccati transform. Mei's breakthrough opened the door for the development of applications such as the finite-difference solution of radiation by a biconical antenna, and the

finite-element solution of scattering by two-dimensional inhomogeneous dielectric cylinders. [Refs. 1 and 2]

Attempting to improve upon these applications, Morgan, Chang and Mei [Ref. 7] conducted research with the goal of extending the unimoment technique to problems involving three-dimensional, axisymmetric inhomogeneous structures. During the initial period of this research, another major breakthrough occurred. This breakthrough was the discovery of the coupled azimuthal potential (CAP) formulation [Ref. 7]. The CAP formulation became the basis for construction of several numerical algorithms. These algorithms incorporated the capability to perform scattering computations for inhomogenous axisymmetric metallic and penetrable structures [Ref. 2].

Utilizing a triregional unimoment method, these numerical algorithms were incorporated into a versatile finite-element computer program. The triregional unimoment method was chosen because of its ability to minimize the bandwidth of the global system matrix, and its ability to minimize the number of nodes in the finite-element mesh. The finite-element computer program generates scattering solutions for an arbitrarily shaped inhomogeneous lossy penetrable body of revolution being impinged upon by multiple incident fields. [Ref. 2]

B. COUPLED AZIMUTHAL POTENTIALS

The solution of electromagnetic field problems involving generalized axisymmetric penetrable structures using the classical TE and TM potential formulations are restricted to a handful of configurations, and have at most inhomogeneities in one-dimension. By utilizing the coupled azimuthal potential (CAP) formulation, solutions are valid in lossy inhomogeneous isotropic rotationally symmetric media, as shown in Figure 2-1, where the complex material parameters are invariant to the azimuthal coordinate. [Ref. 1]

Utilizing normalized circular cylindrical coordinates, as also shown in Figure 2-1, the first step in developing the necessary equations is to decompose the total electromagnetic field into azimuthal modes through the use of an exponential Fourier series,

$$\bar{E}(R,Z,\Phi) = \sum_{m=-\infty}^{\infty} \bar{e}_m(R,Z)\exp(jm\Phi) \tag{2.1}$$

and

$$\eta_0\bar{H}(R,Z,\Phi) = \sum_{m=-\infty}^{\infty} \bar{h}_m(R,Z)\exp(jm\Phi) \tag{2.2}$$

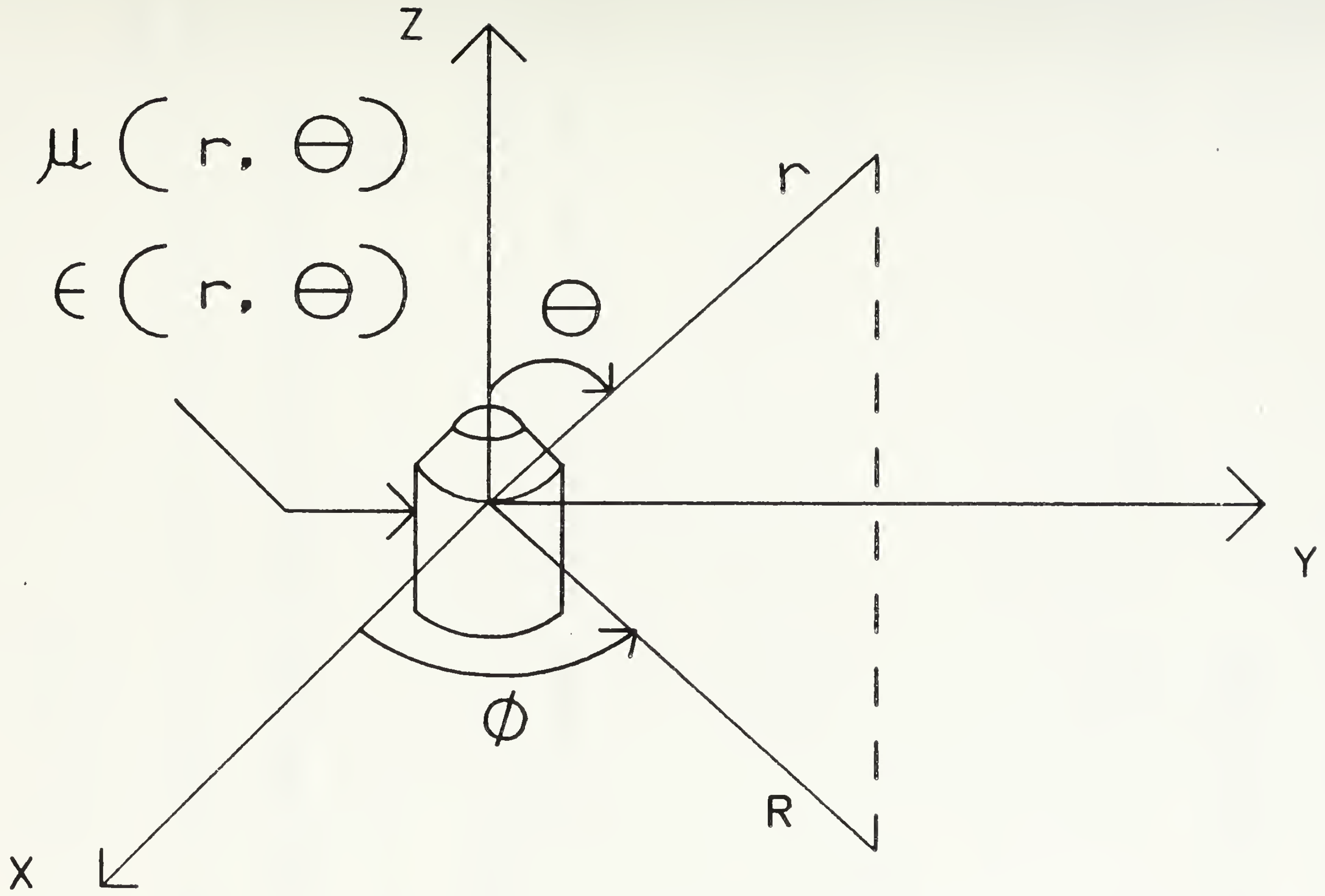


FIG. 2-1. AXIALLY-SYMMETRIC MEDIUM IN NORMALIZED CIRCULAR CYLINDRICAL COORDINATES

where

$$(R, Z, \Phi) = (k_0 \rho, k_0 z, \Phi) \tag{2.3}$$

with

$$k_0 = \frac{2\pi}{\lambda_0} \tag{2.4}$$

and

$$\eta_0 = 120\pi \ \Omega \quad . \tag{2.5}$$

Two uniformly continuous coupled azimuthal potential functions

$$\psi_1(R, Z, m) \ \text{ and } \ \psi_2(R, Z, m)$$

can be used to generate the modal electromagnetic field vectors,

$$\bar{e}_m(R,Z) = jf_m(m\nabla\psi_1 + R\mu_r\hat{\Phi} \times \nabla\psi_2) + \hat{\Phi}\frac{\psi_1}{R} \qquad (2.6)$$

$$\bar{h}_m(R,Z) = jf_m(m\nabla\psi_2 - R\varepsilon_r\hat{\Phi} \times \nabla\psi_1) + \hat{\Phi}\frac{\psi_2}{R} \quad , \qquad (2.7)$$

where the two-dimensional gradient operator is

$$\nabla = \hat{R}\frac{\partial}{\partial R} + \hat{Z}\frac{\partial}{\partial Z} \qquad (2.8)$$

while the multiplicative function is given by [Refs. 1, 2, and 7]

$$f_m(R,Z) = [\mu_r(R,Z)\varepsilon_r(R,Z)R^2 - m^2]^{-1} \quad . \qquad (2.9)$$

For a two-dimensional cut in the (R,Z) plane of Figure 2-1, the coupled azimuthal potential functions satisfy a formally self-adjoint system of second-order partial differential equations. Utilizing either Euler-Lagrange or stationary-theorem approaches, it can be shown that there is an easily managed variational principle which can be applied in addition to the partial differential equation formulation.

This variational formulation is based upon the existence of a stationary function which manifests itself as a surface integral over the planar cross section, S,

$$F = \int\int_S L(R,Z,\psi_1,\psi_2,\nabla\psi_1,\nabla\psi_2)\ dR\ dZ \qquad (2.10)$$

where the Lagrangian L, which is unique to within an arbitrary constant multiplier and arbitrary independent additive function, is of the form [Refs. 1 and 2]

$$L = f_m \left[\nabla\psi_1 \cdot (R\varepsilon_r \nabla\psi_1 + m\hat{\Phi} \times \nabla\psi_2) + \right.$$

$$\left. \nabla\psi_2 \cdot (R\mu_r \nabla\psi_2 - m\hat{\Phi} \times \nabla\psi_1) \right] -$$

$$\frac{1}{R} \left[\varepsilon_r \psi_1^2 + \mu_r \psi_2^2 \right] \qquad . \qquad (2.11)$$

C. FINITE-ELEMENT ALGORITHM

Having presented an overview of the coupled azimuthal potential formulation, it is now appropriate to explain its application to the electromagnetic scattering problem. Figure 2-2 illustrates the finite-element mesh geometry employed by the numerical algorithm. The (R,Z) planar configuration shown is named the semi-annular conformal (SAC) mesh [Ref. 1]. This mesh is comprised of numerous linear triangular elements, which conform to the surface of the axisymmetric scattering body. This surface conformation simplifies analysis of normal derivative discontinuities at the surface boundary. The semi-annular conformal mesh provides a distinct computational advantage to those

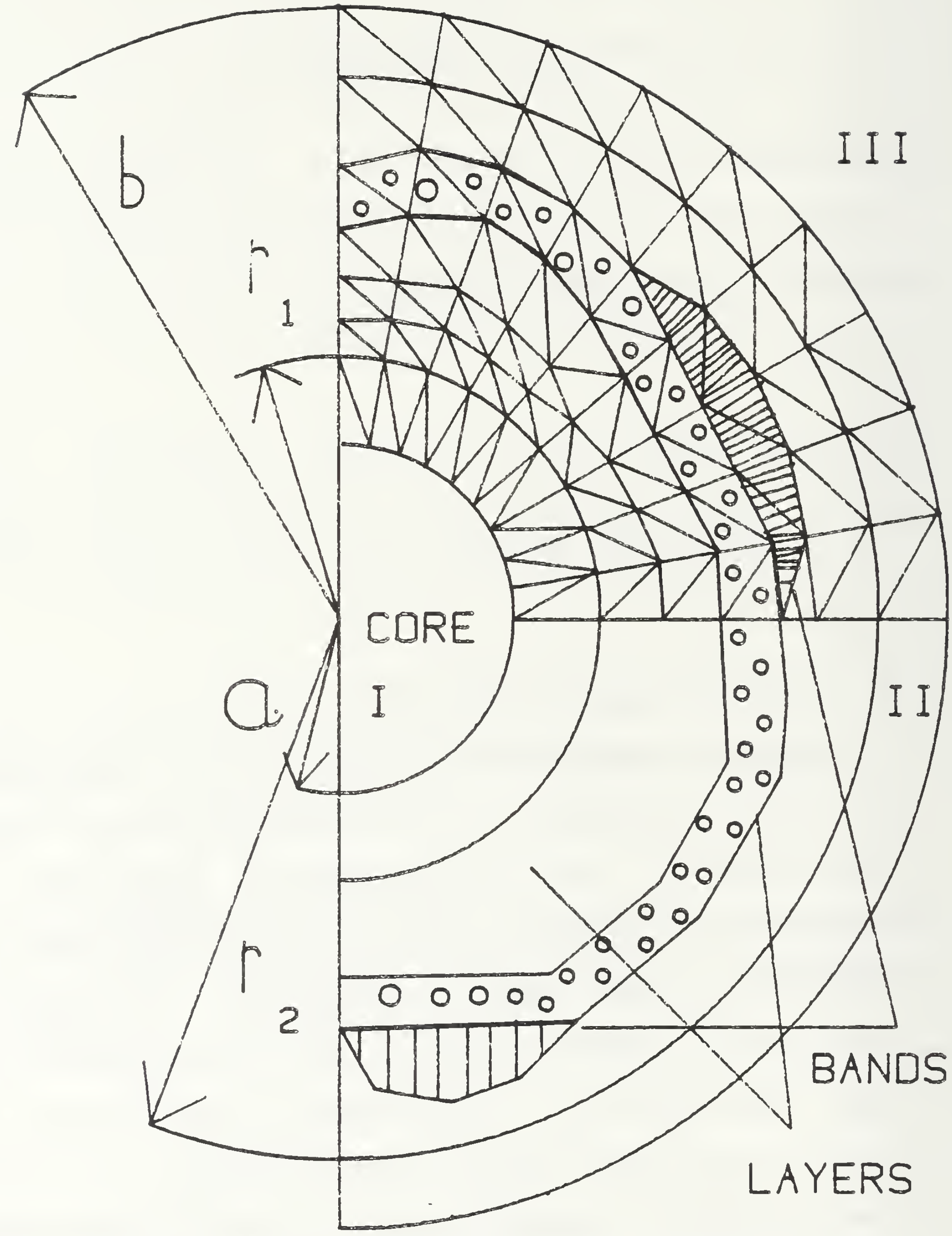


FIG. 2-2. SEMI-ANNULAR CONFORMAL FINITE-ELEMENT MESH FOR AN INHOMOGENEOUS BODY OF REVOLUTION

scattering structures which can support a fairly large homogeneous spherical core [Ref. 2].

Figure 2-2 graphically demonstrates the regions of interest when calculating the solution of the scattering problem. The homogeneous spherical core occupies Region I, the inhomogeneous cross sectional surface of the axisymmetric penetrable scattering body is in Region II, and an unbounded exterior region is in Region III. The surface of interest is shown with the capability of being both banded and layered. The layers are depicted as "films" that run the entire length of the surface, while bands are depicted as isolated "pockets". The possibility exists, as shown, that all the bands and all the layers have differing material parameters, thus establishing a structure that is truly inhomogeneous. Note that the surfaces of Region I and Region III bound the surface of interest located in Region II. These surfaces, $r = a$ and $r = b$, can therefore be used to specify Dirichlet boundary conditions [Ref. 2]. The numerical solution for the electromagnetic fields in the finite-element domain (Region II) can be computed for each of a set of boundary conditions. By requiring continuity of the electromagnetic fields at the interfaces where $r = r_1$ and $r = r_2$, the unknown expansion coefficients for the core and scattered fields can be obtained [Ref. 1]. This is the essential concept behind the unimoment method.

In Region II of Figure 2-2, the semi-annular conformal mesh is composed of nodes, with each interior node surrounded by six triangular regions, known as "elements". The azimuthal field components of the mesh are described by

$$\hat{\Phi} \cdot \bar{e}_m(R,Z) = \sum_{n=1}^{N} e_m(n) u_n(R,Z) \tag{2.12}$$

and

$$\hat{\Phi} \cdot \bar{h}_m(R,Z) = \sum_{n=1}^{N} h_m(n) u_n(R,Z) \tag{2.13}$$

where $u_n(R,Z)$ represents the area coordinate linear pyramid basis functions. Each of these basis functions has linear variation in the six triangular elements around each node, while maintaining a value of one at the central node and a value of zero at all bounding nodes. Each set of $e_m(n)$ and $h_m(n)$ represent one of the N complex nodal values found in the azimuthal field. To initiate the variational finite-element technique, the linear shape-function expansions in Equations 2.12 and 2.13 are substituted into the surface integral stationary function F described in Equation 2.10. Imposing the null derivative condition with respect to each

of the unknown coordinates representing the interior nodal fields, while constraining the coordinates of the boundary nodal fields, allows the computation of the stationary point of the function F. The resulting linear system for the nodal fields is described by

$$\sum_{i=1}^{N} \left\{ e_m(i) \int\int_{S_n} \left[f_m R\varepsilon_r \nabla(Ru_n) \cdot \nabla(Ru_i) - R\varepsilon_r u_n u_i \right] dR\, dZ \right.$$

$$\left. + h_m(i) \int\int_{S_n} \left[m f_m \nabla(Ru_n) \cdot \hat{\Phi} \times \nabla(Ru_i) \right] dR\, dZ \right\} = 0 \qquad (2.14)$$

and

$$\sum_{i=1}^{N} \left\{ h_m(i) \int\int_{S_n} \left[f_m R\mu_r (Ru_n) \cdot \nabla(Ru_i) - R\mu_r u_n u_i \right] dR\, dZ \right.$$

$$\left. - e_m(i) \int\int_{S_n} \left[m f_m \nabla(Ru_n) \cdot \hat{\Phi} \times \nabla(Ru_i) \right] dR\, dZ \right\} = 0 \qquad (2.15)$$

where n = each internal node. When these equations are combined with the semi-annular conformal finite element mesh

analysis, each equation, for a given interior node n, will relate $e_m(n)$ and $h_m(n)$ to 14 adjacent node azimuthal field values $e_m(i)$ and $h_m(i)$. Some of these adjacent field values may be specified on a boundary. The result of conducting these calculations for each value of n is a sparse global system matrix, containing complex elements and displaying a diagonal tri-block submatrix structure. Applying the Dirichlet boundary conditions to the semi-annular conformal finite-element mesh shown in Figure 2-2, the numerical solution for the electromagnetic fields in Region II is obtained by utilizing a two-sweep block-by-block matrix inversion algorithm related to the Riccati transform for difference equations. The diagonal tri-block submatrix structure is given by

$$\underset{\sim}{A}_k \underset{\sim}{\Psi}_{k-1} + \underset{\sim}{B}_k \underset{\sim}{\Psi}_k + \underset{\sim}{C}_k \underset{\sim}{\Psi}_{k+1} = \underset{\sim}{\Phi}_k \quad . \qquad (2.16)$$

A_k, B_k, and C_k are sparse submatrices which constitute the tri-block diagonal structure of the global system matrix. Φ_k is generated from Equations 2.14 and 2.15 through the use of the Dirichlet boundary conditions. [Ref. 2]

The mesh construction and matrix generation are conducted on a local basis in a sequential manner. The analysis starts at the positive Z-axis, and works clockwise in equal θ-increments. Using a simple minimum distortion scheme at each step, the mesh is constructed to conform to multiple approximate piecewise smooth surfaces. The Riccati transform sparse matrix inversion technique conducts a "first sweep" and a "backsweep". On the "first sweep", the values of R_k and S_k are calculated, where

$$\underset{\sim}{R}_{k+1} = -(\underset{\sim}{B}_k + \underset{\sim}{A}_k \cdot \underset{\sim}{R}_k)^{-1} \cdot \underset{\sim}{C}_k \tag{2.17}$$

$$\underset{\sim}{S}_{k+1} = (\underset{\sim}{B}_k + \underset{\sim}{A}_k \cdot \underset{\sim}{R}_k)^{-1} \cdot (\underset{\sim}{\Phi}_k - \underset{\sim}{A}_k \underset{\sim}{S}_k) \quad . \tag{2.18}$$

On the "backsweep", the values of the solution vector are obtained, where the solution vector is given by [Refs. 1 and 2]

$$\underset{\sim}{\Psi}_{k-1} = \underset{\sim}{R}_k \underset{\sim}{\Psi}_k + \underset{\sim}{S}_k \quad . \tag{2.19}$$

A principal advantage of this numerical method is that the Riccati transform technique provides a systematic procedure. This procedure divides a larger area into several smaller areas defined by the tri-block matrix substructure, existing between any three adjacent radial rows of nodal values. Analysis using this type of smaller area procedure allows all operations including mesh construction, matrix element generation and matrix inversion to be performed at the same time. [Ref. 2]

D. UNIMOMENT METHOD

The unimoment method is a computational procedure which allows numerical values to be calculated for the complex coefficients of the incident, scattered and spherical core electromagnetic field expansions when dealing with the semi-annular conformal finite-element mesh. The scattering solution is obtained by representing the electromagnetic fields of Region I and Region III, as shown in Figure 2-2, in terms of TE and TM spherical harmonic expansions, with unknown core and scattered field coefficients. Using superposition of boundary conditions and interior solutions for the azimuthal fields, the core expansion is established as a boundary condition along $r = a$ while the scattered field expansion added to the incident field is established as a boundary condition along $r = b$. The two existing sets of boundary conditions are modal spherical harmonic core fields along $r = a$, with zero fields along $r = b$, and

spherical harmonic and incident fields at r = b, with null fields at r = a. The multiple boundary conditions will generate multiple solutions at the nodes where $r = r_1$ or $r = r_2$, (i.e.,along the boundaries). The method used to obtain the expansion coefficients sets the weighted superposition of interior solutions to the appropriate original field expansions. Along the boundaries at $r = r_1$ and $r = r_2$, this method enforces equality utilizing least squares analysis. [Refs. 1 and 2]

This method of analysis is only valid if the scattered electromagnetic fields everywhere outside of the interior mesh region can be represented by field harmonics. These representations appear as infinite series, and must be mathematically truncated to facilitate numerical calculations. Decomposing the plane-wave incident fields into azimuthal modes yields

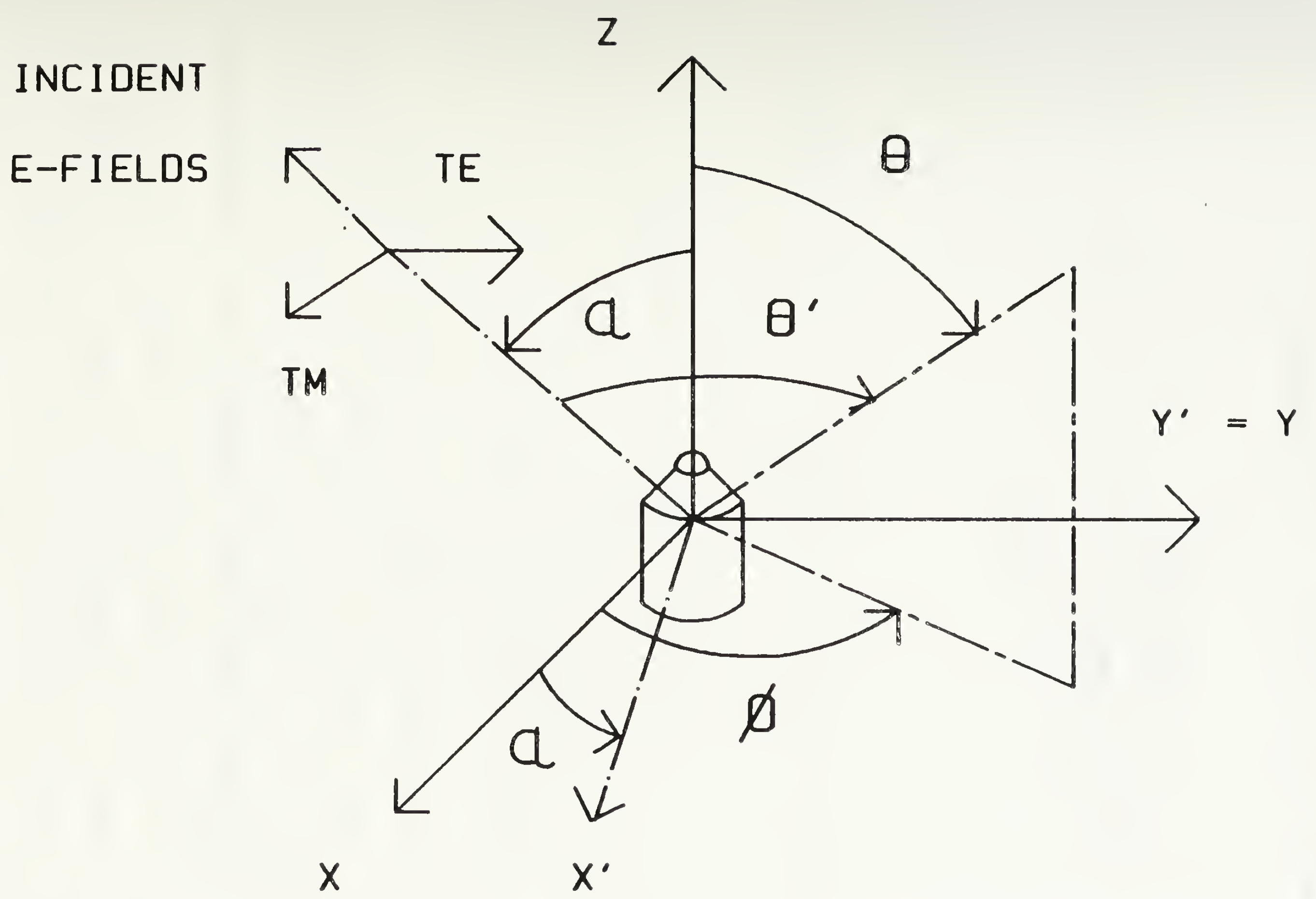
$$E_{\Phi}(r,\theta,\Phi) = \sum_{m=0}^{M} \varepsilon_m e_{m,\Phi}(r,\theta) \begin{bmatrix} j\sin(m\Phi) \\ \cos(m\Phi) \end{bmatrix} \tag{2.20}$$

$$\eta_0 H_{\Phi}(r,\theta,\Phi) = \sum_{m=0}^{M} \varepsilon_m h_{m,\Phi}(r,\theta) \begin{bmatrix} \cos(m\Phi) \\ j\sin(m\Phi) \end{bmatrix} \tag{2.21}$$

where $\epsilon_m = 1$ if $m = 0$, $\epsilon_m = 2$ if $m \geq 1$; M is the truncation index, and the polarization convention (TE/TM) ,depicted in Figure 2-3, is used in the bracketed sinusoids. [Ref. 2]

During computations, the numerical scattering solution algorithm is run concurrently with the sequential Riccati transform finite-element solution in Region II. On the forward sweep, boundary conditions for both the incident and modal fields are generated at each node row being used in that particular sweep step. On the backsweep, boundary conditions along the $r = r_1$ and $r = r_2$ contours are used to compute each nodal value. The incident field and spherical harmonic field for each node is formed, and used to generate the nodal residual fields. During the backsweep, inner product integrations are formed, and these inner product results are added to the field moment matrix, **T**, and the incident field driving vector, D corresponding to each incident field. The scattering solution is obtained by multiplying the inverse of the field-moment matrix with the driving vector to obtain the field expansion coefficients. The complex scattering amplitude $f(\theta,\Phi)$ is defined in terms of the far-zone E-field,

$$\bar{E}(r,\theta,\Phi) = \bar{f}(\theta,\Phi)\,\frac{\exp(-jk_0 r)}{k_0 r} \;;\quad r \gg \frac{d_{max}}{\lambda_0} \qquad (2.22)$$



(X, Y, Z) COORDINATES WITH RESPECT TO BODY OF REVOLUTION
(X′, Y′, Z′) COORDINATES WITH RESPECT TO INCIDENT FIELD

FIG. 2-3. POLARIZATION CONVENTION

where d_{max} is the maximum scattering body dimension and where the components of $f(\Theta,\Phi)$ are given by

$$f_{\Phi}(\theta,\Phi) = \sum_{m=0}^{M} \varepsilon_m \begin{bmatrix} j\sin(m\Phi) \\ \cos(m\Phi) \end{bmatrix} \sum_{\substack{n=m \\ n\neq 0}}^{N_m^{III}} a_{m,n}^{III}\, j^{n+1}$$

$$\cdot\, D_\theta P_n^m + b_{m,n}^{III}\, j^n \frac{m P_n^m}{\sin(\theta)} \qquad (2.23)$$

$$f_{\theta}(\theta,\Phi) = \sum_{m=0}^{M} \varepsilon_m \begin{bmatrix} \cos(m\Phi) \\ j\sin(m\Phi) \end{bmatrix} \sum_{\substack{n=m \\ n\neq 0}}^{N_m^{III}} a_{m,n}^{III}\, j^{n}$$

$$\cdot\, \frac{m P_n^m}{\sin(\theta)} - b_{m,n}^{III}\, j^{n+1} D_\theta P_n^m \quad . \qquad (2.24)$$

The scattering cross section is found to be [Refs. 1 and 2]

$$\sigma_s = \frac{\lambda_0^2}{2\pi|\bar{E}^i|^2} \sum_{m=0}^{M} \varepsilon_m \sum_{\substack{n=m \\ n\neq 0}}^{N_m^{III}} \frac{2(n+1)\cdot(n+m)!}{(n+1)\cdot(n-m)!}$$

$$\cdot \left\{ \left|a_{m,n}^{III}\right|^2 + \left|b_{m,n}^{III}\right| \right\} \quad . \qquad (2.25)$$

Utilizing the theoretical principles and mathematical calculations just discussed, Morgan [Ref. 1] wrote the computer program EMSCAT to compute electromagnetic field scattering from layered bodies of revolution. Extensive interactive computer aided design features have been added to EMSCAT by Morgan and Connolly. These computer-aided design features, driving the change of name from EMSCAT to EMCAD, allow structures composed of multiple dielectric materials to be graphically constructed or modified by the user, and then evaluated by EMCAD for scattering. The remainder of this thesis is concerned with the utilization

and application of EMCAD, as well as with modifications and improvements of this transformation.

III. EMCAD INPUT DATA CONSTRUCTION

As has been previously explained, EMCAD computes electromagnetic field scattering from penetrable bodies of revolution using the coupled azimuthal potential formulation in conjunction with a variational finite-element technique and a tri-regional unimoment method. The numerical solution is performed using a two-sweep Riccati transform with circumferential marching. In order to provide relatively "fast", consistently accurate answers, yet be flexible enough to allow the user to vary any of a number of input parameters as conditions warrant, EMCAD requires sufficient input data to adequately depict the penetrable body of revolution, as well as the desired incident and scattered field orientations of the user. This accuracy and flexibility is accomplished through the utilization of keyboard input and/or mouse input to form both material and structure input data files. Mouse input deals with the graphical interface, and as such will be discussed in Chapter IV.

A. KEYBOARD INPUT

Information obtained by keyboard input is fairly diversified in nature. This data ranges from input and output file names, to desired mesh parameters, from selection of phi angles, to selection of incidence angles.

To facilitate ease in understanding and running the program, EMCAD allows a user to input the necessary information according to the level of expertise needed. For the "Novice" level, the program gives brief explanations of input values, describes the required format, and gives typical values where applicable. For the other option, the "Expert" level assumes the user is familiar with the input parameters and formats, and simply prompts for the required inputs. A more detailed explanation of each facet of the keyboard input is contained in Appendix A, EMCAD USER'S MANUAL.

B. DATA FILE INPUT

Once the information obtained by keyboard input is established, the remaining data becomes that which describes the penetrable body of revolution being considered. This descriptive data can be separated into two distinct areas of information. The first area of information is the data which conveys the material composition of each layer of the structure. Call this the "material file". The second area of information is the data which defines the structural shape of each layer of the penetrable body, referred to as the "structur file". As shown in Figure 3-1, there are three methods of data construction which can be used to generate the "Material" file and the "Structur" file. The quotation marks indicate that the data files can be named at the user's discretion. Method 2 and Method 3 utilize

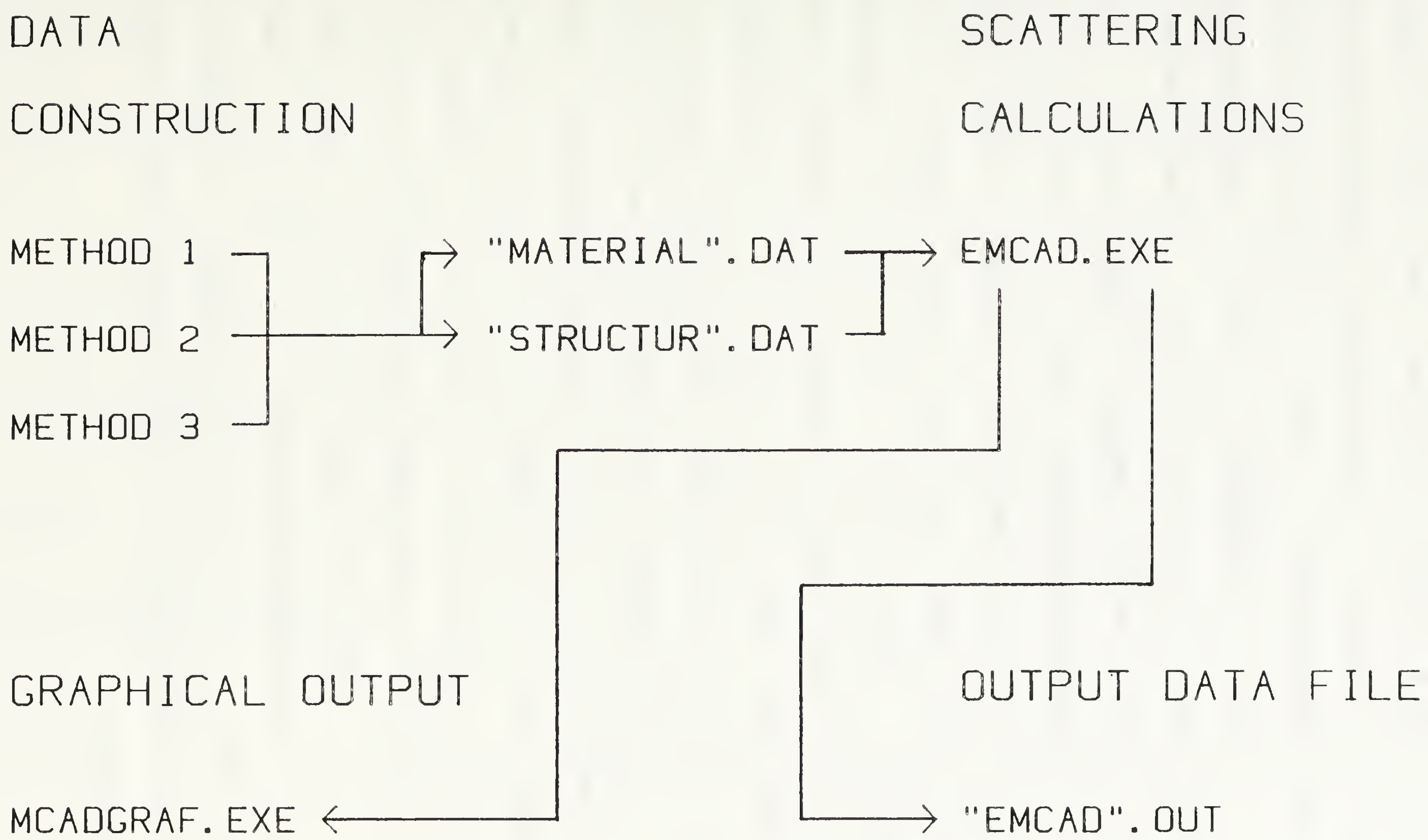


FIG. 3-1. SCATTERING CALCULATION INPUT-OUTPUT RELATIONSHIP

computer aided design features, and will therefore be discussed in Chapter IV, Interactive Graphics Features.

Data construction by way of Method 1 is depicted in Figure 3-2 and is composed of special case programs which generate data files for specific geometrically shaped objects. For example, program SPHERE allows the user to define a layered penetrable sphere, offset from center if desired, with as many penetrable layers as can be handled by EMCAD at this time. The user enters material dielectric parameters, layer radii, and required resolution as prompted, and the program writes the necessary "Material" and "Structur" data into files in the required EMCAD format. CONE and CYLINDER were written specifically for EMCAD. As with SPHERE, they allow the user to establish the body of revolution as a multilayered structure. Source code for SPHERE, CONE, and CYLINDER can be found in Appendices J, K, and L respectively. Whenever a computer program generates data and data files, the capability exists for the data to be generated manually, or by any other means available, and then entered into the appropriate data files manually through the use of the keyboard. Keyboard entry is shown as a viable method of data construction in Method One, Two and Three.

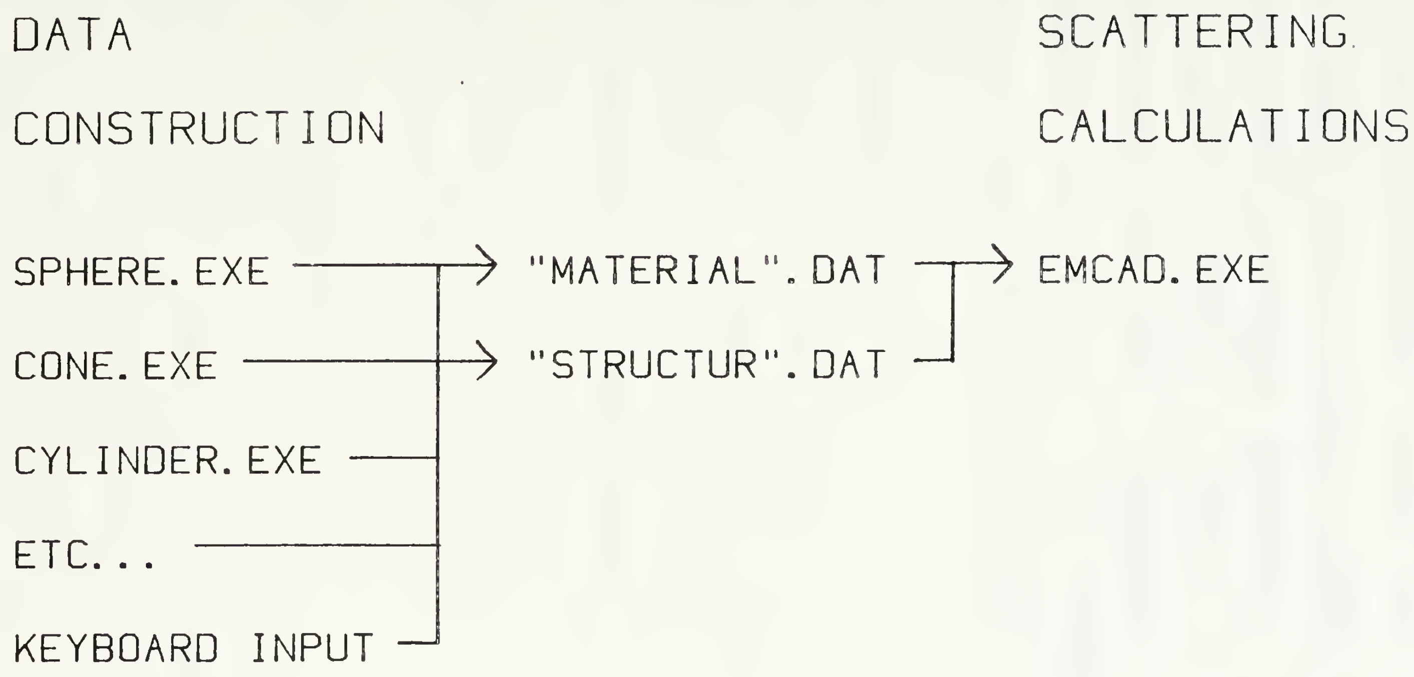


FIG. 3-2. SCATTERING CALCULATION METHOD NUMBER ONE

Spheres, cones, and cylinders are simple geometric shapes, and are extremely useful for elementary applications; however, nature is complex, and not always made up of simple geometry and elementary shapes. To obtain realistic results, realistic penetrable bodies are needed. It would indeed be tedious to attempt to write a structure defining computer program for each new shape encountered. As such, the optimum solution indicates the need for the implementation of computer aided graphics.

IV. INTERACTIVE GRAPHICS FEATURES

Interactive graphics simply refers to the ability to take data at hand, and depict that data in some visual form, where a graph is the most commonly utilized depiction. A special subset of interactive graphics, known as computer aided design, also allows the user to draw a picture on the screen, and convert that picture to data.

EMCAD was written in the programming language of Ryan-McFarland FORTRAN (Appendix M). Microcompatibles GRAFMATIC (Appendix M), and later Jewell Technologies GRAFPLUS (Appendix M) were chosen as the commercially available screen graphics and hard copy support for EMCAD output graphics. Because the output format that was used readily allowed conversion to an acceptable format, West Coast Consultants CURVE DIGITIZER (Appendix M) was chosen as the computer aided design package.

A. COMPUTER AIDED DESIGN

There are three means of constructing the data for penetrable bodies of revolution, as previously shown in Figure 3-1. Method 1 has already been discussed, and is the predecessor to Method 2. In Method 2, the "Material" data can be generated by one of the three programs written specifically for EMCAD, or by direct keyboard input into a data file if desired. The change from Method 1 occurs in

the construction of the structure defining data. As shown in Figure 4-1, the "Curveout" data is obtained by using CURVE DIGITIZER. In simple terms, this means that the user draws the picture on the screen using a mouse with the computer aided design package. Once the user is satisfied with the single, or multilayered drawing on the screen, the data is saved in the format established by the computer aided design package. An example of the two-dimensional cross section of a three-dimensional body of revolution which was designed using CURVE DIGITIZER is shown in Figure 4-2.

The data file that is saved is a valid description of the structure, but is not in a form which can be used by EMCAD. This format conversion is accomplished by a program written specifically for this purpose, EMCADIN. The computer aided design output data is written in X,Y format, with special codes interspersed throughout. EMCAD requires radius format and a normalizing factor of 2π must be taken into account. The output data is not necessarily evenly spaced, whereas EMCAD requires evenly spaced input data. It is conceivable that the data is of insufficient resolution for use by EMCAD. Primitive data should be of high resolution (i.e., 361 points). The program EMCADIN rectifies all of these problems, as well as computes the minimum and maximum radius values required. The output of

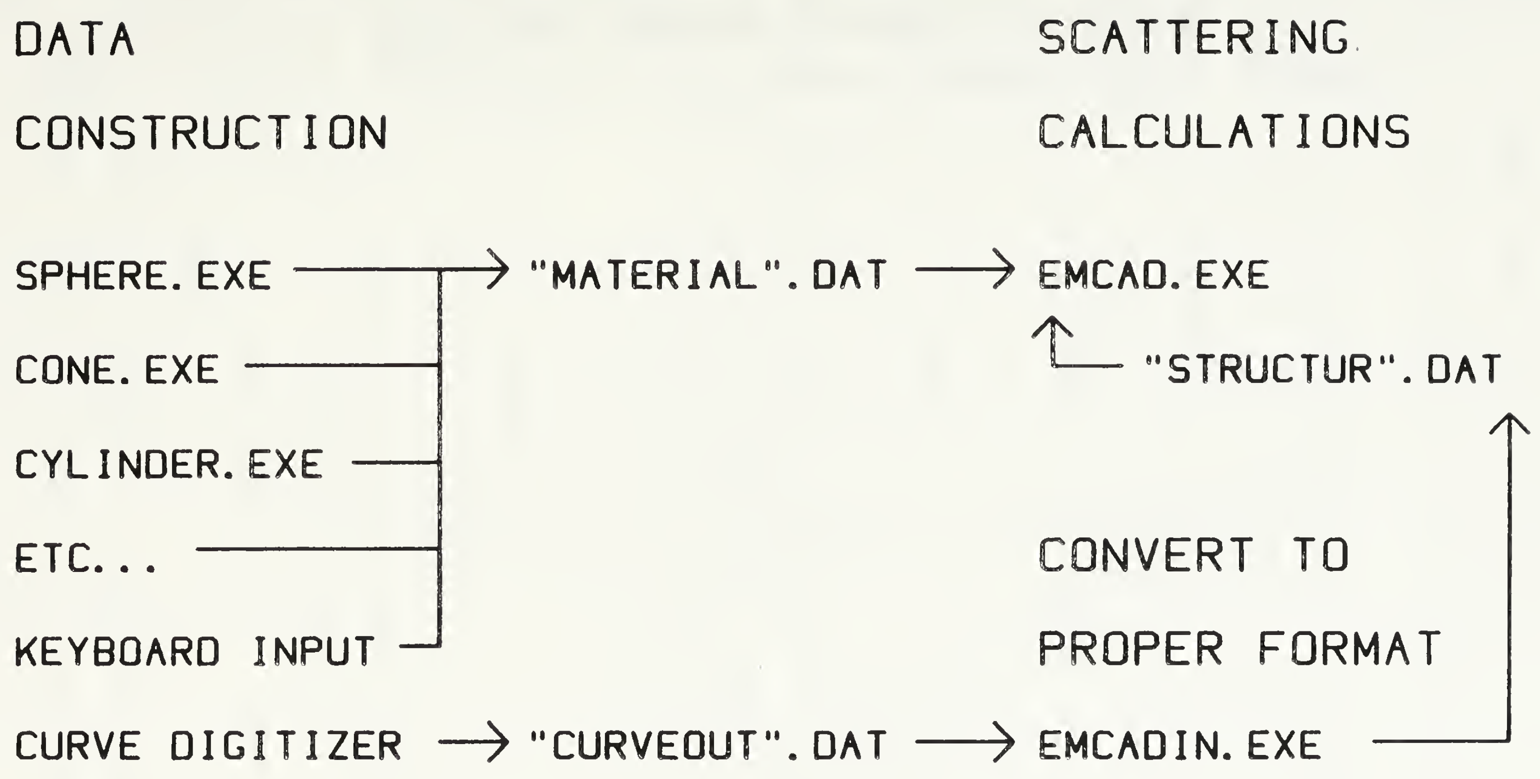


FIG. 4-1. SCATTERING CALCULATION METHOD NUMBER TWO

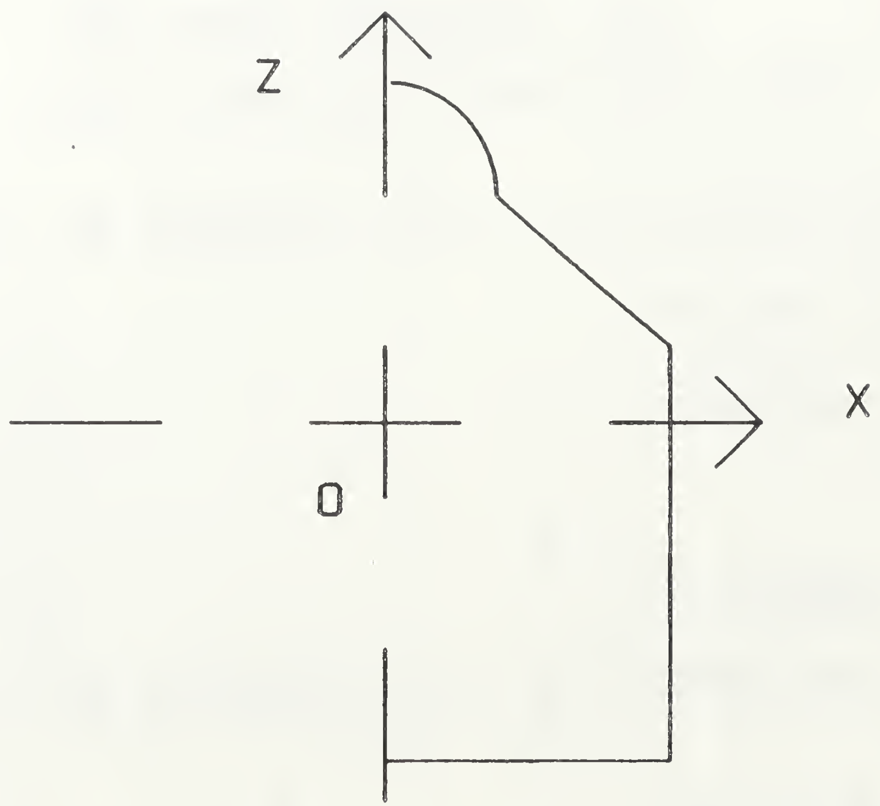


FIG. 4-2. EXAMPLE OF TWO-DIMENSIONAL CROSS SECTION OF BODY OF REVOLUTION DESIGNED USING CURVE DIGITIZER

EMCADIN is the data file "Structur". A more detailed description can be found in Appendix D, EMCADIN USER'S MANUAL.

Method 3, as shown in Figure 4-3, is actually a union of Method 1 and Method 2. "Material" data is generated as described in both previous methods. Structural data is generated as in Method 1, but is then imported to the computer aided design package for modifications. The data to be modified must be changed by CRVDIGIN, written especially for this purpose, into a format which can be accepted as input by CURVE DIGITIZER. CRVDIGIN is explained in greater detail in Appendix F, CRVDIGIN USER'S MANUAL. Once the data has been read into the computer aided design package, the remainder of the development is essentially Method 2.

B. INTERACTIVE OUTPUT GRAPHICS

Having constructed the "Material" and the "Structur" input data files using one of the three methods discussed, it is now possible to conduct the scattering calculations. Figure 4-4 shows that the execution of EMCAD yields output data and output graphics.

As also shown in Figure 4-4, EMCAD generates four data files which are used as input to the graphics program MCADGRAF. It is significant to note that the extensions of these filenames have special meaning. TEP refers to TE incidence (Figure 2-3), F-Phi; TET refers to TE incidence,

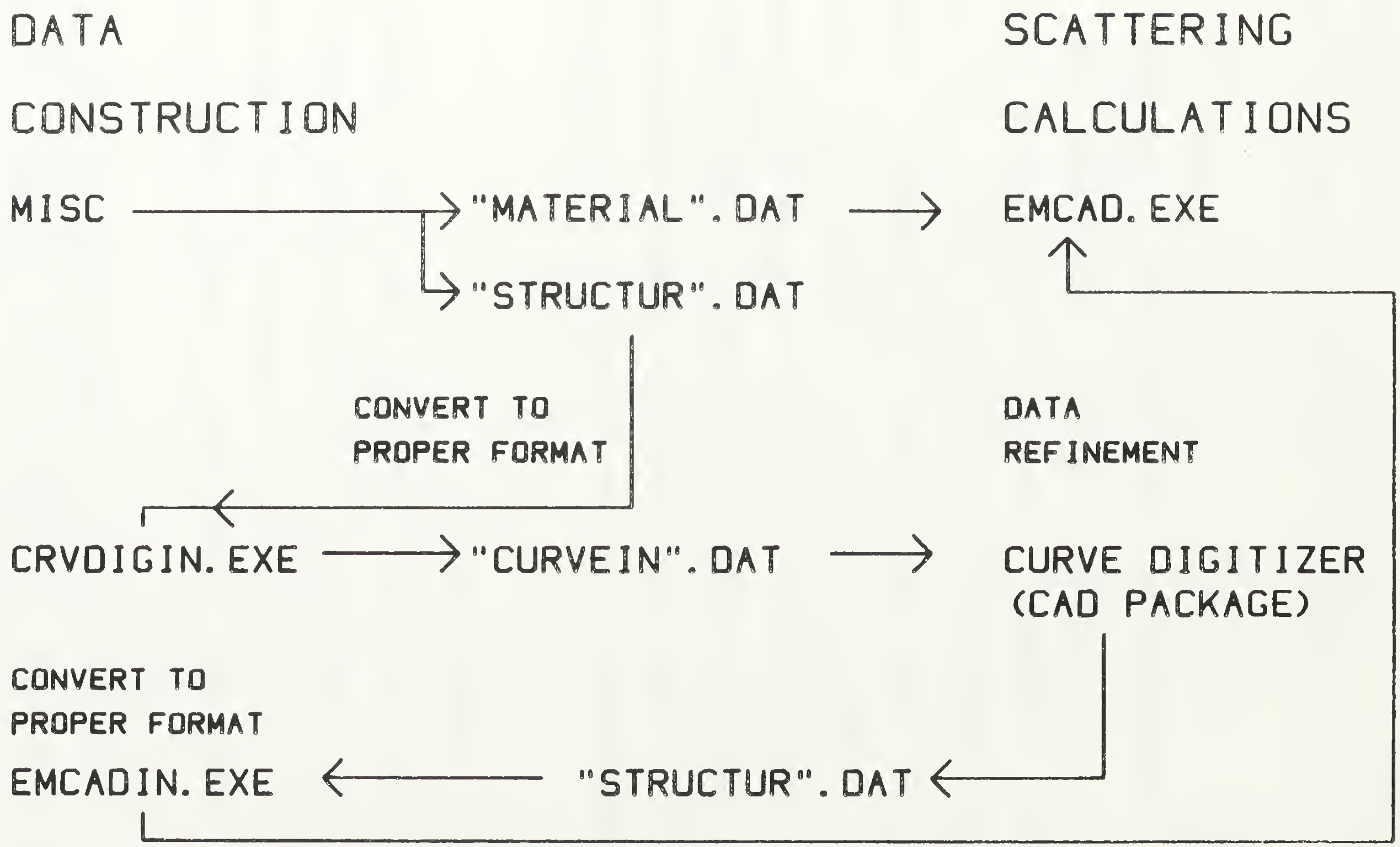


FIG. 4-3. SCATTERING CALCULATION METHOD NUMBER THREE

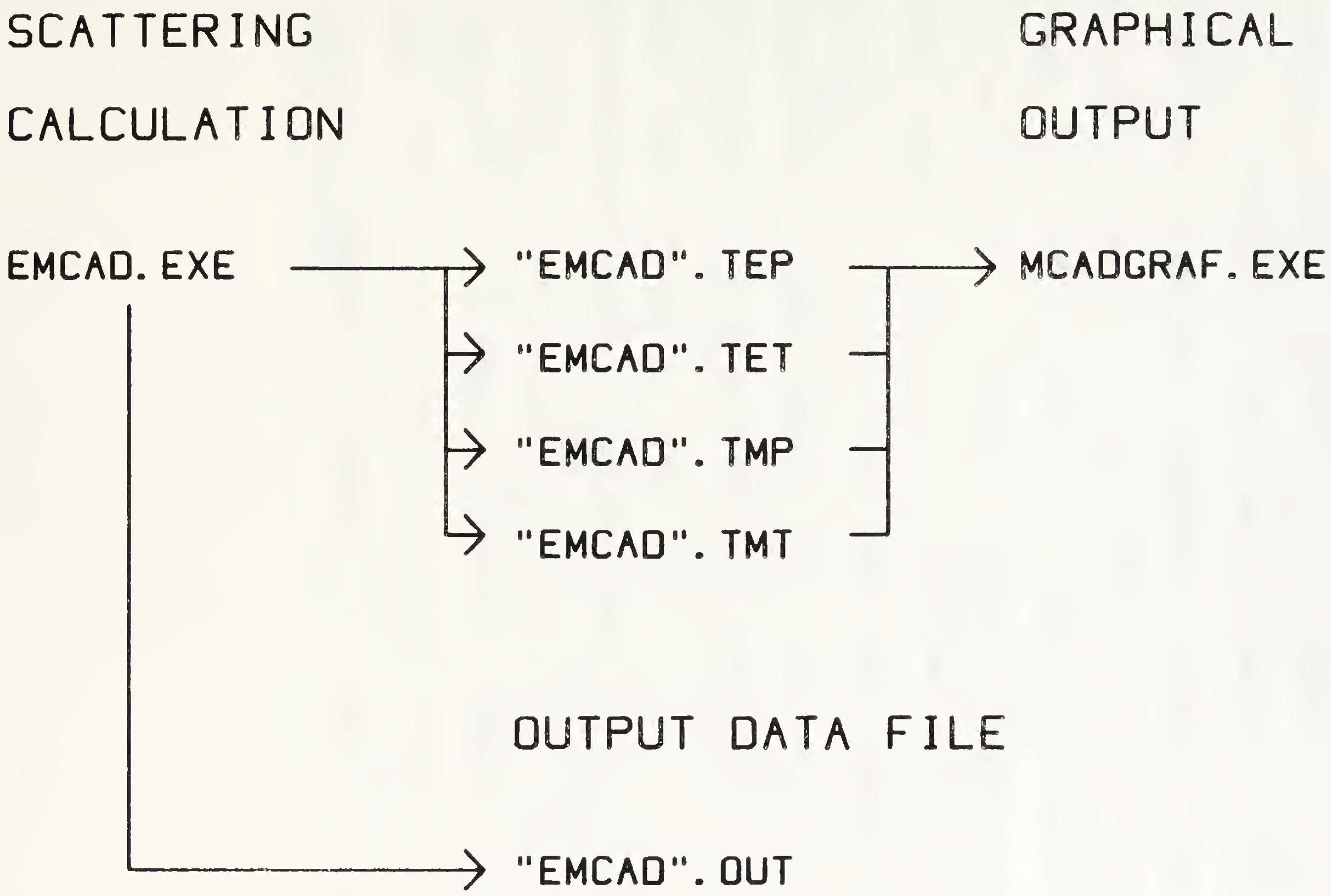


FIG. 4-4. SCATTERING CALCULATION OUTPUT AND GRAPHICS

F-Theta; TMP refers to TM incidence (Figure 2-3), F-Phi; TMT refers to TM incidence, F-Theta.

MCADGRAF is a menu driven output graphics program. Appendix H, MCADGRAF USER'S MANUAL, gives examples of all the different input menus. The menu structure gives the user some general information concerning the use of the program, as well as user choice of aspect ratio, data file type selection, data file name input, incidence angle selection, and phi angle selection. Once these selections have been made, MCADGRAF produces four output graph types, giving the user the capability to make a printout of any or all of these. The first graph type is magnitude versus theta in a rectangular coordinate system, as shown in Figure 4-5. The second graph type is phase versus theta in a rectangular coordinate system, as shown in Figure 4-6. The third graph type is magnitude versus theta in a polar coordinate system, shown in Figure 4-7. The final type is simply an axis rotated version of the third type, for a different perspective, as shown in Figure 4-8.

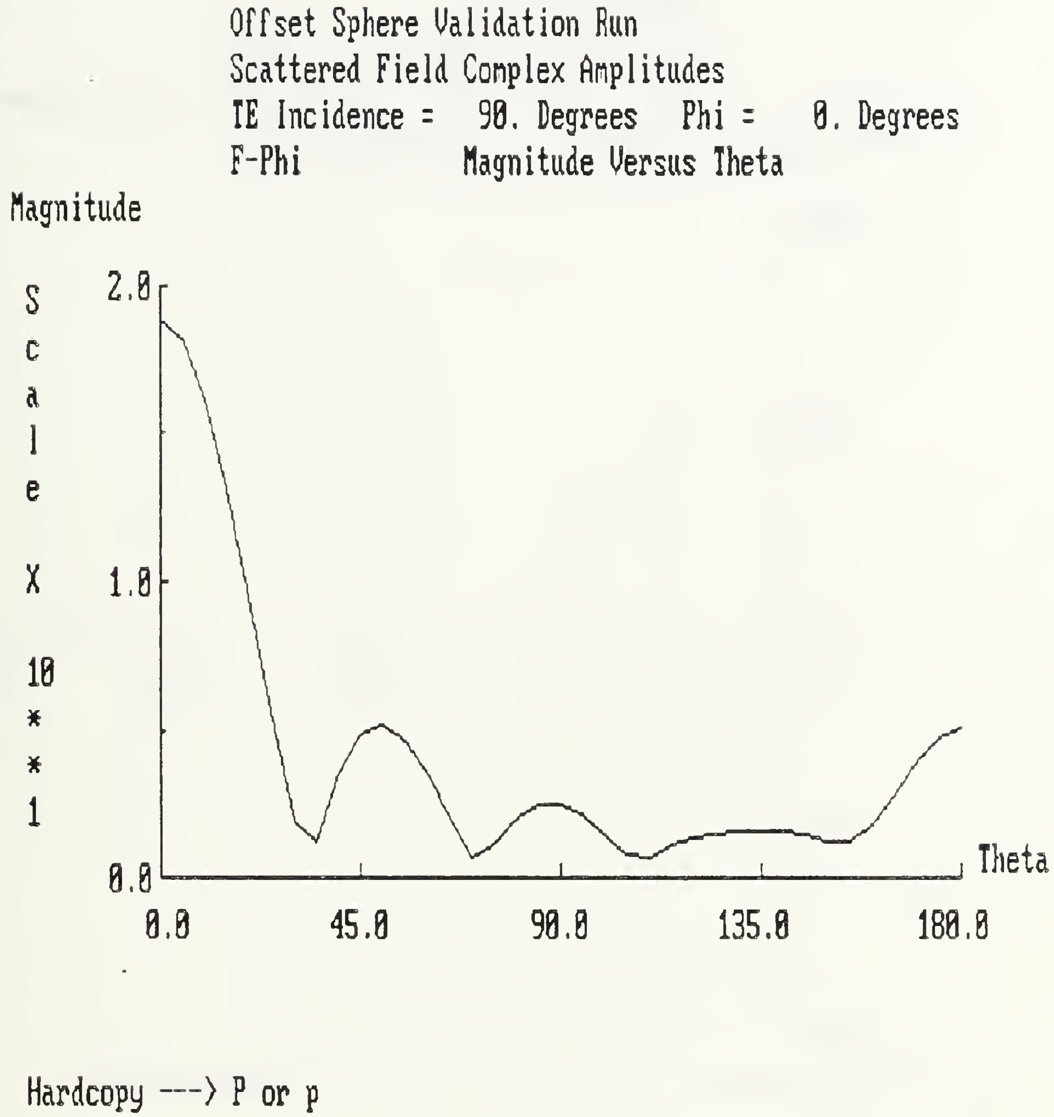


FIG. 4-5. MCADGRAF GRAPHICAL OUTPUT

MAGNITUDE VERSUS THETA

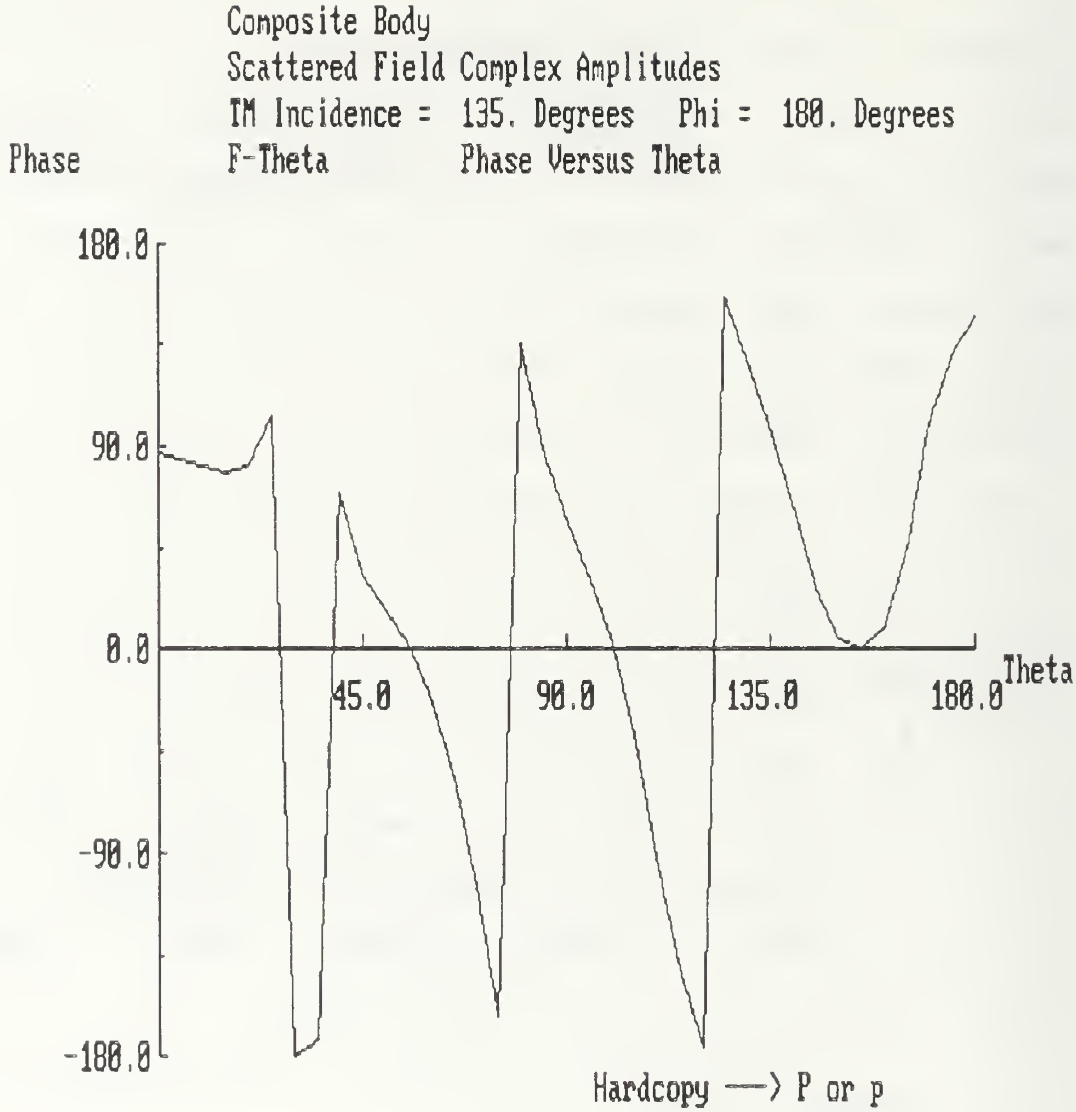


FIG. 4-6. MCADGRAF GRAPHICAL OUTPUT

PHASE VERSUS THETA

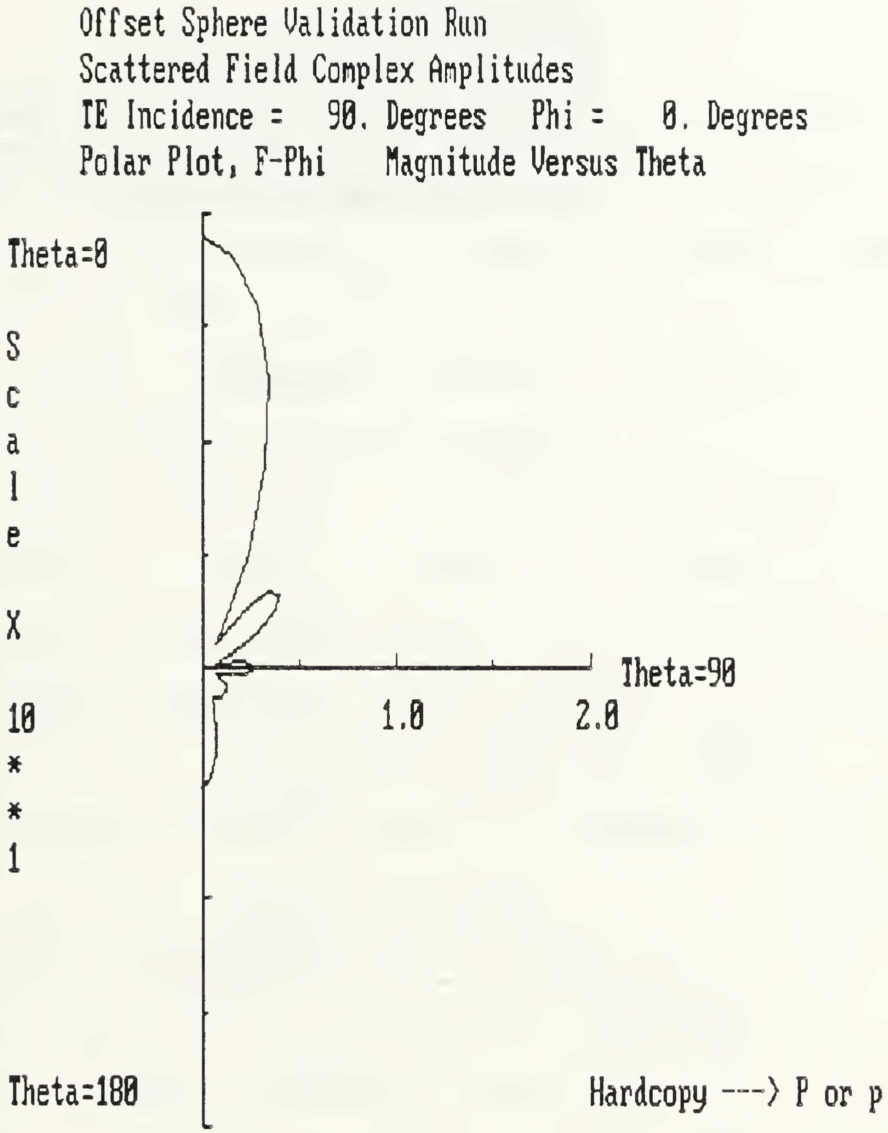


FIG. 4-7. MCADGRAF GRAPHICAL OUTPUT

MAGNITUDE VERSUS THETA, POLAR PLOT

Composite Body
Scattered Field Complex Amplitudes
TM Incidence = 135. Degrees Phi = 180. Degrees
Polar Plot, F-Theta Magnitude Versus Theta
Axis Rotated Minus Ninety Degrees

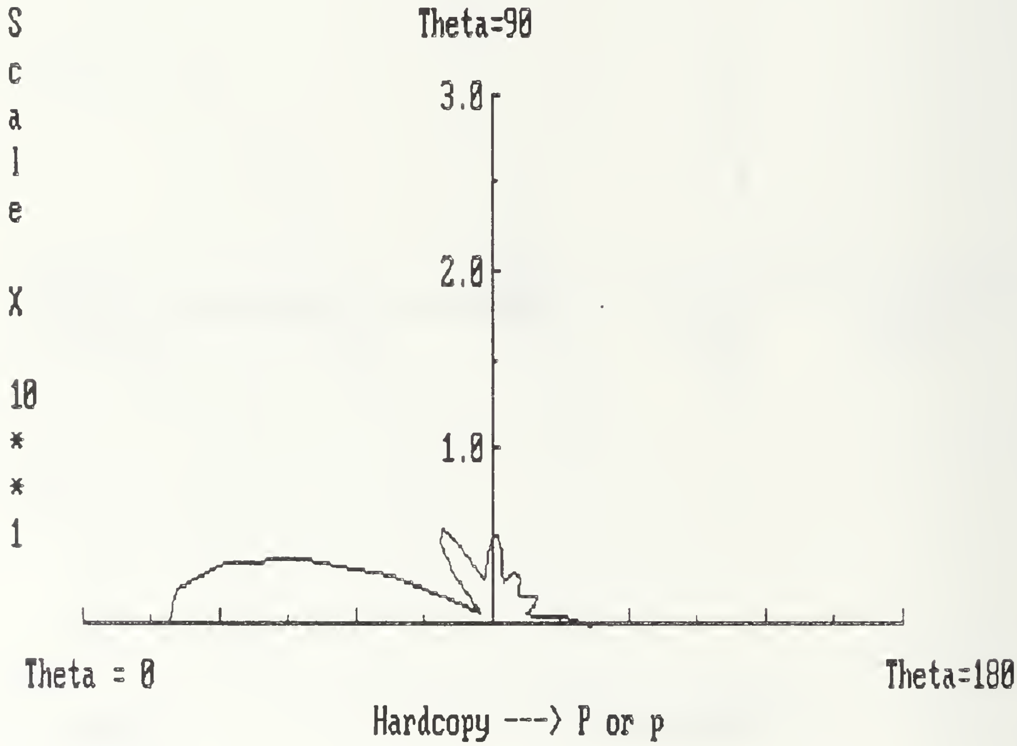


FIG. 4-8. MCADGRAF GRAPHICAL OUTPUT
MAGNITUDE VERSUS THETA, POLAR PLOT
AXIS ROTATED MINUS NINETY DEGREES

V. VALIDATION

In order to transform the existing program EMSCAT to the modified version entitled EMCAD, it was imperative to ensure that both the accuracy and the reliability of EMCAD were at least as good as that of EMSCAT. It was also essential to establish flawless compatibility between EMCAD and any program written to support EMCAD.

The initial step used to validate the operation of EMCAD consisted of a comparison between the results obtained from EMCAD and the exact theoretical values of a Mie series far-field complex amplitude E-plane pattern for an offset lossless (ϵ_r = 2.6) penetrable dielectric sphere. Run parameters chosen included broadside incidence (incidence angle = 90 degrees), sphere radius equal to 0.8 (as a factor of wavelength), and z-axis offset = 0.16 (as a factor of wavelength). The series truncations at M=7, $N_M{}^I$=8 and $N_M{}^{II}$=15 were chosen for the variable mesh scheme. These values were the same as those used by Morgan [Ref.1] in his initial validation of EMSCAT. The results of this comparison are shown in Figure 5-1 for a phi angle of 0 degrees, and in Figure 5-2 for a phi angle of 90 degrees.

Having demonstrated the close resemblance between the results obtained from EMCAD, and those predicted by theory, the next step used to validate EMCAD required a comparison

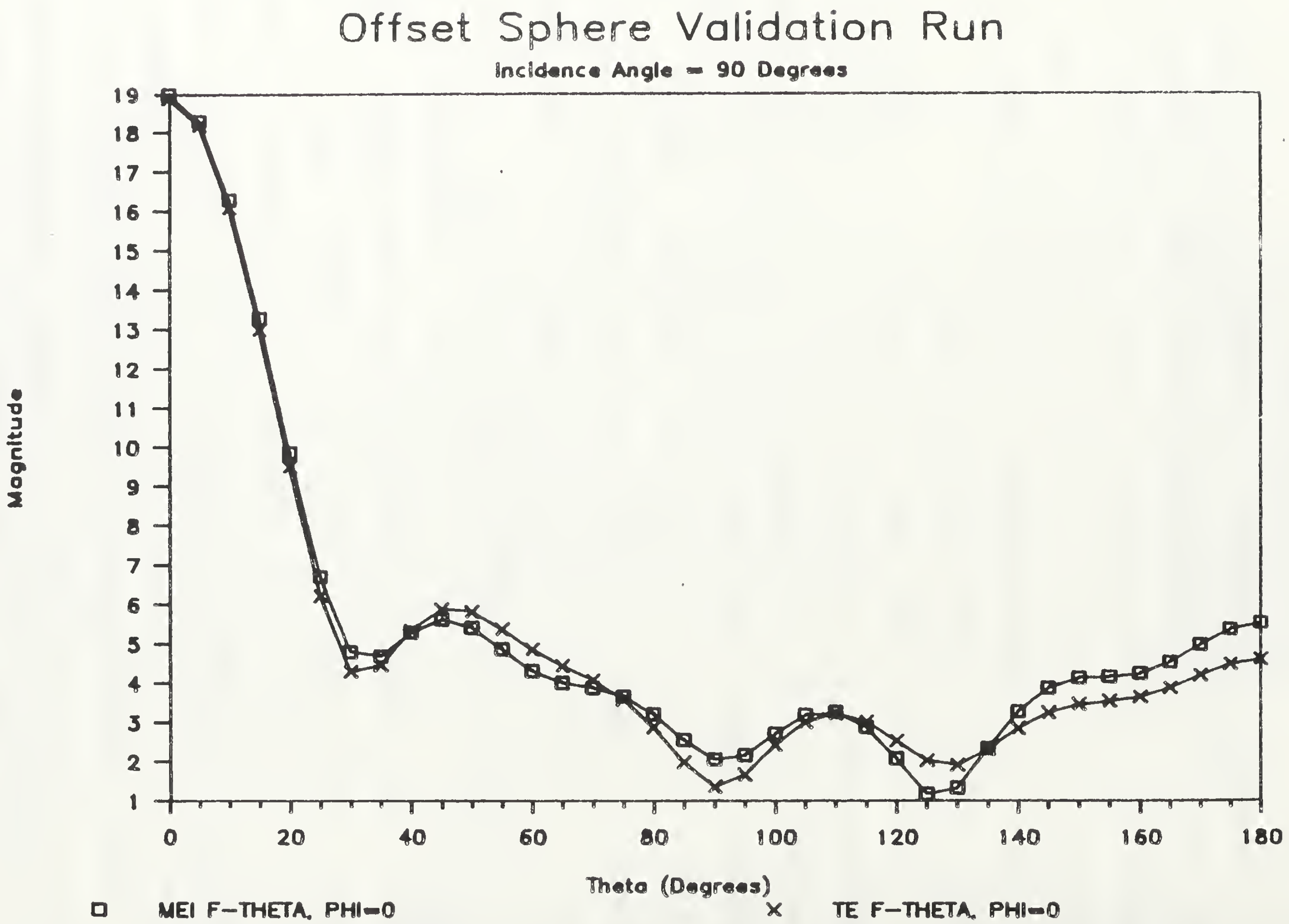


FIG. 5-1. OFFSET SPHERE SCATTERING (#1)

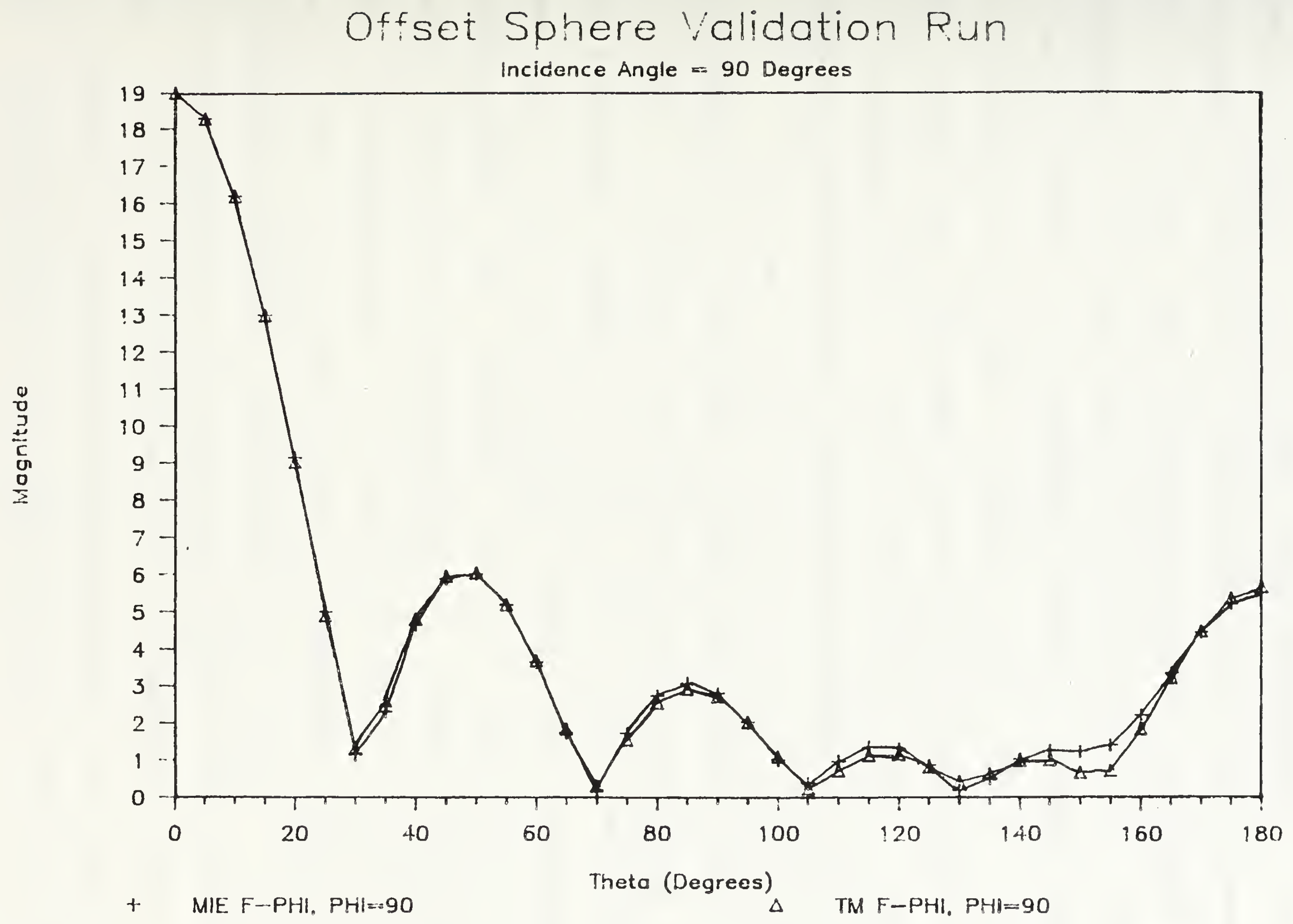


FIG. 5-2. OFFSET SPHERE SCATTERING (#2)

between the results obtained from EMCAD and actual measured data. Morgan [Ref. 1] validated the original version of EMSCAT with extensive experimental verification. Having the results of this experimental verification readily available presented the unique opportunity to validate the operation of EMCAD with the same experimental data used to validate EMSCAT. For each axisymmetric body constructed, parameters chosen included TE and TM incidence with incidence angles of 0, 45, 90, 135 and 180 degrees. The series truncations at $M=9$, $N_M^I=15$ and $N_M^{II}=20$ were chosen for the variable mesh scheme. All experimental bodies were constructed of plexiglass (polymethylmethacrylate) with material parameters defined by $\epsilon_r = 2.59 - j0.017$. [Refs. 1 and 2]

The decision was made to demonstrate the versatility and reliability of the program by utilizing a number of significant shapes which could be readily fabricated for experimental measurements. Since a sphere had already been used, and shown to work, a bicone was chosen as an appropriate figure. A representation of the solid bicone utilized is depicted in Figure 5-3. The comparison for the solid bicone validation is shown in Figure 5-4. The hollow bicone of Figure 5-5 demonstrates increasing complexity. The sphere shown inside the bicone is a free space cavity in the object, although it could be made of any material. The comparison for the hollow bicone is presented in Figure 5-6. The next figure chosen was that of a hollow cylinder. The

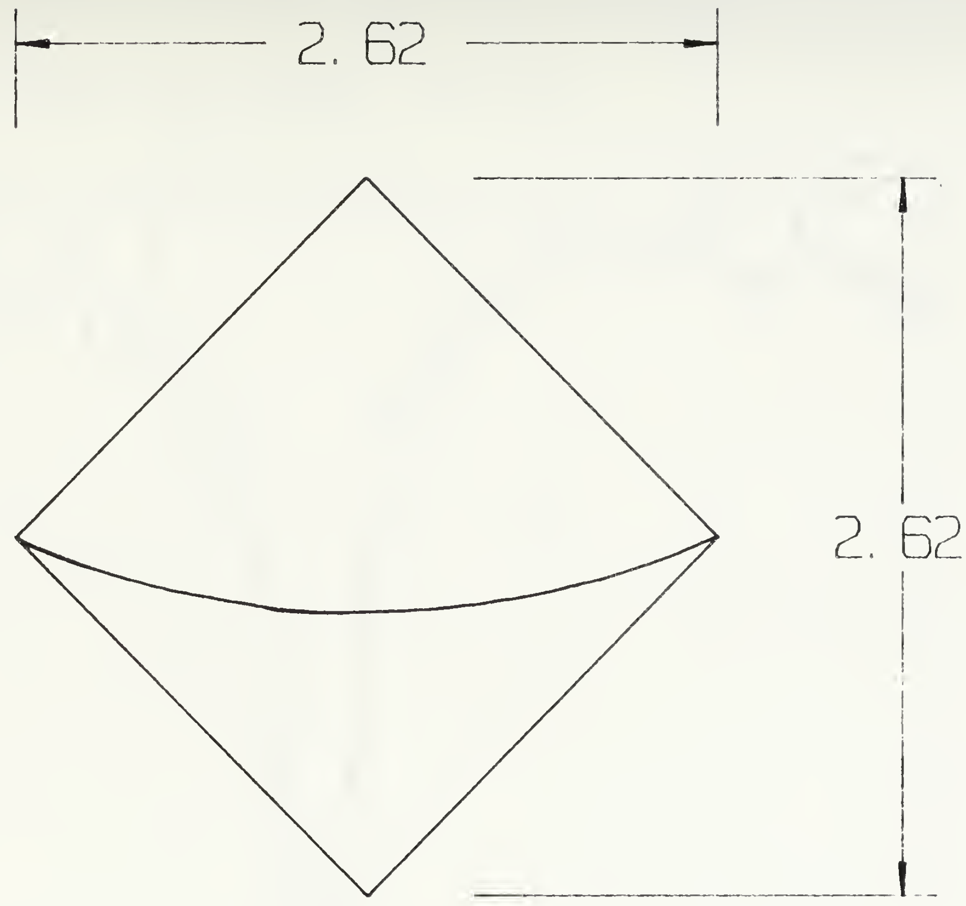


FIG. 5-3. SOLID BICONE USED FOR SCATTERING CALCULATION VALIDATION (COORDINATES IN TERMS OF WAVELENGTH)

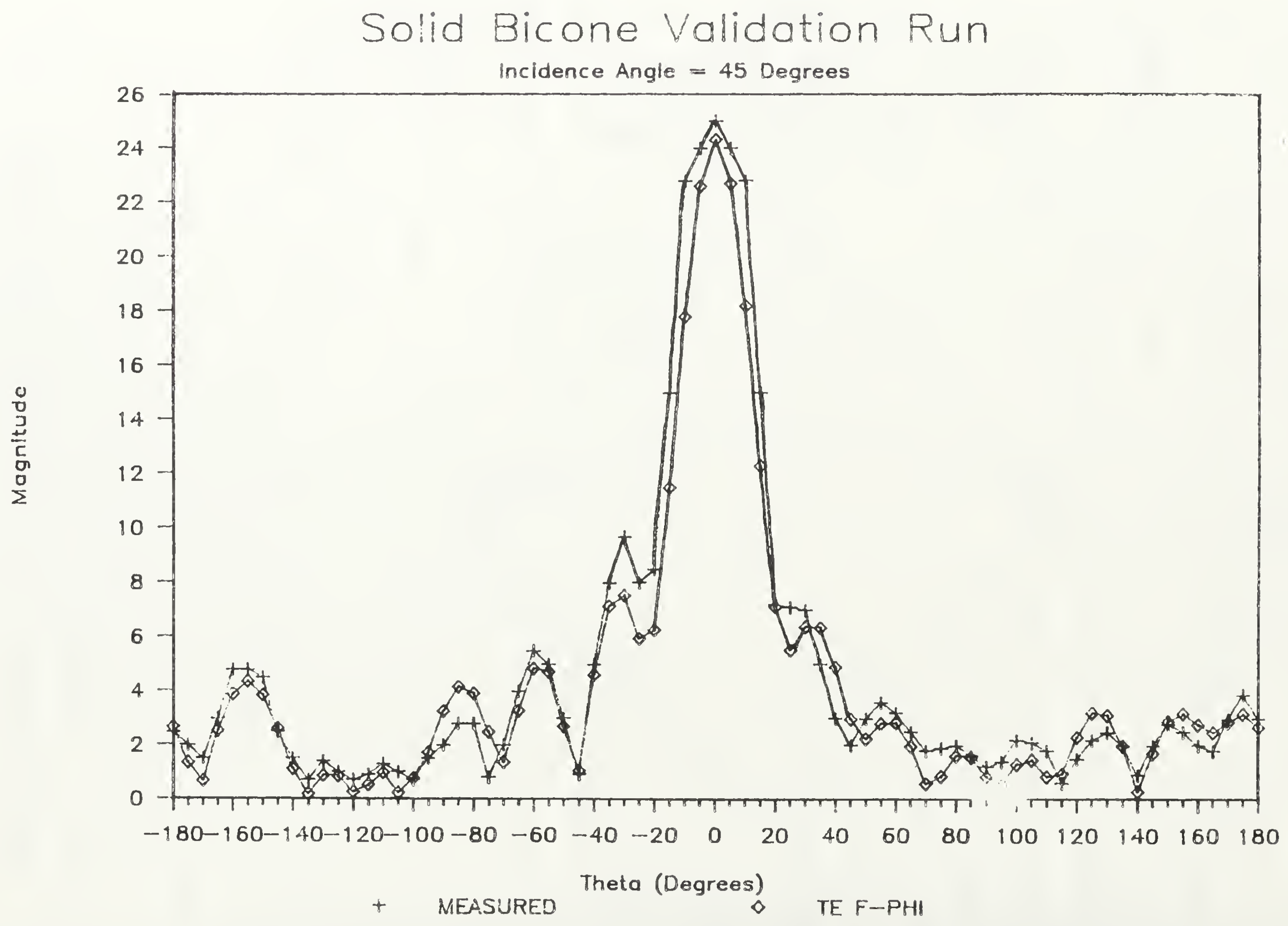


FIG. 5-4. SOLID BICONE SCATTERING

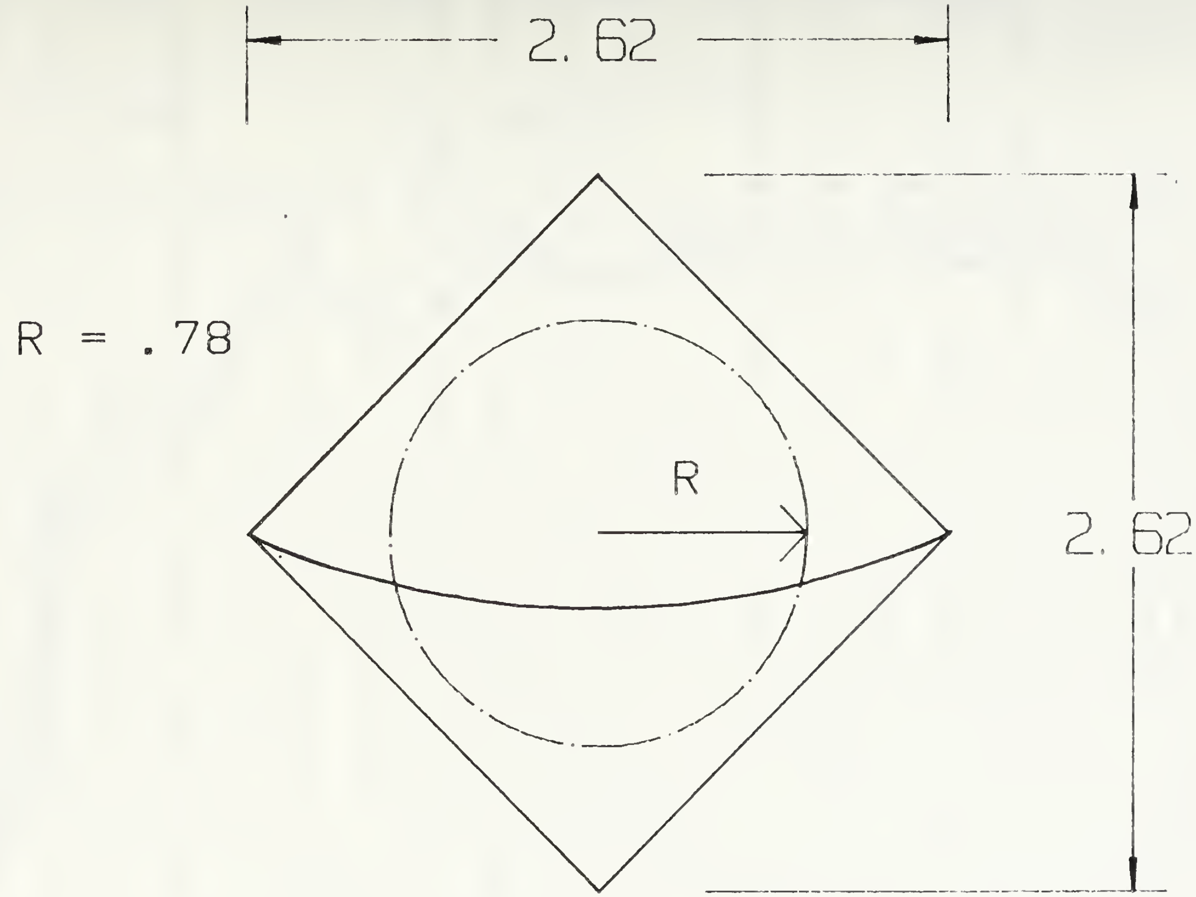


FIG. 5-5. HOLLOW BICONE USED FOR SCATTERING CALCULATION VALIDATION (COORDINATES IN TERMS OF WAVELENGTH)

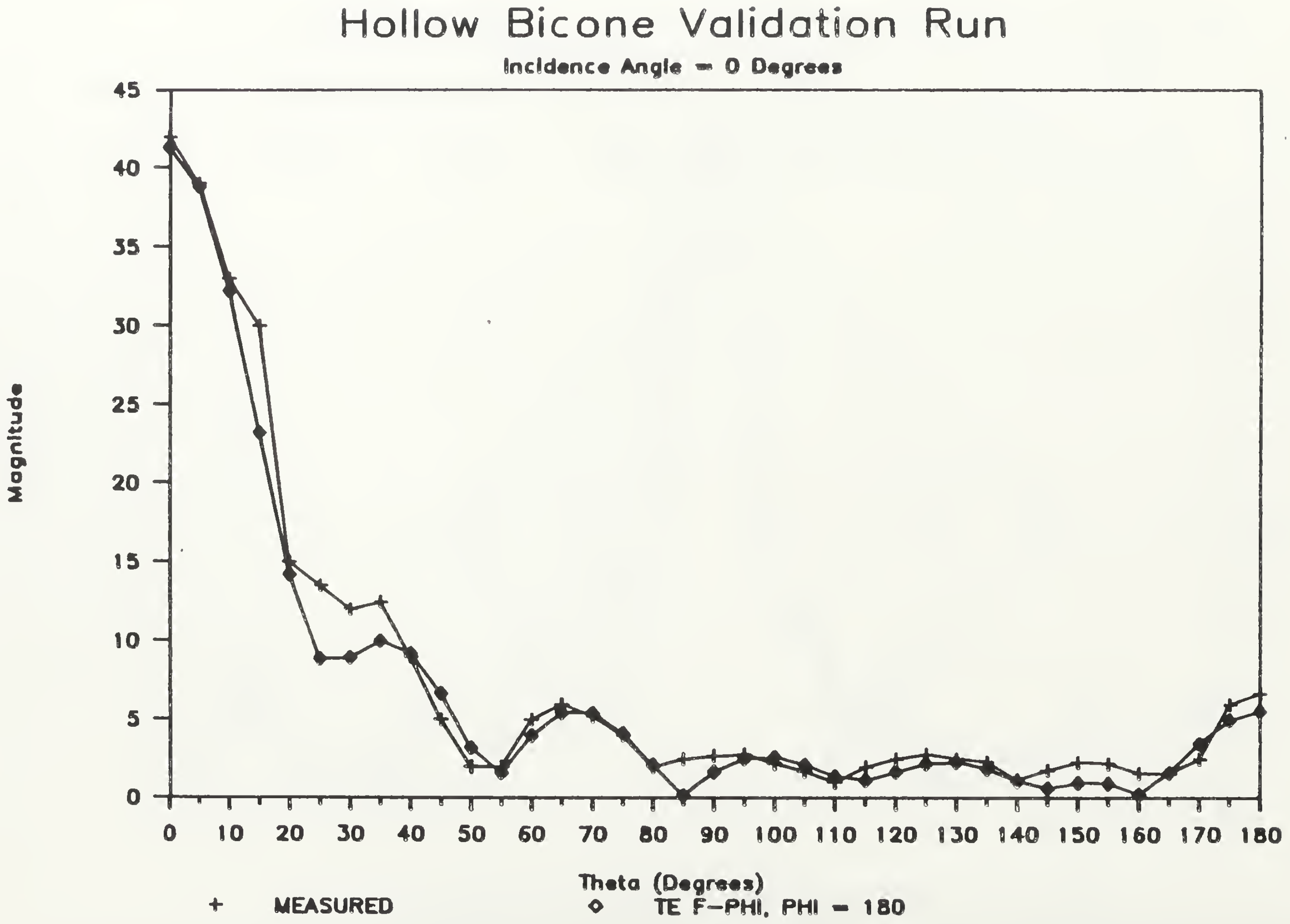


FIG. 5-6. HOLLOW BICONE SCATTERING

internal spherical cavity had the same dimension as that found in the hollow bicone. The height of the cylinder was 1.9 (in terms of wavelength) and the radius of the base of the cylinder was .945 (in terms of wavelength). The validation comparison for a phi angle of 0 degrees is depicted in Figure 5-7 and the validation comparison for a phi angle of 90 degrees is presented in Figure 5-8.

To this point, all validation runs have correlated extremely well with the experimental data, and all experimental figures have consisted of combinations of simple geometric shapes such as spheres, cones, and cylinders. These are extremely useful for elementary applications, however, nature is complex, and not always made up of nice geometry or elementary shapes. To prove that EMCAD could handle more realistic bodies, the composite structure of Figure 5-9 was constructed. Since a program was not written to generate the "structur" data required for EMCAD input, the decision was made to utilize CURVE DIGITIZER (CAD package) to produce the required information. The representation of the composite body of Figure 5-9, required as input for EMCAD, is depicted in Figure 5-10. Note that this representation is composed of data defining a two dimensional axisymmetric body. The actual comparison for this structure is shown in Figure 5-11. As was the case before, the data generated by EMCAD correlates extremely well with the experimental data.

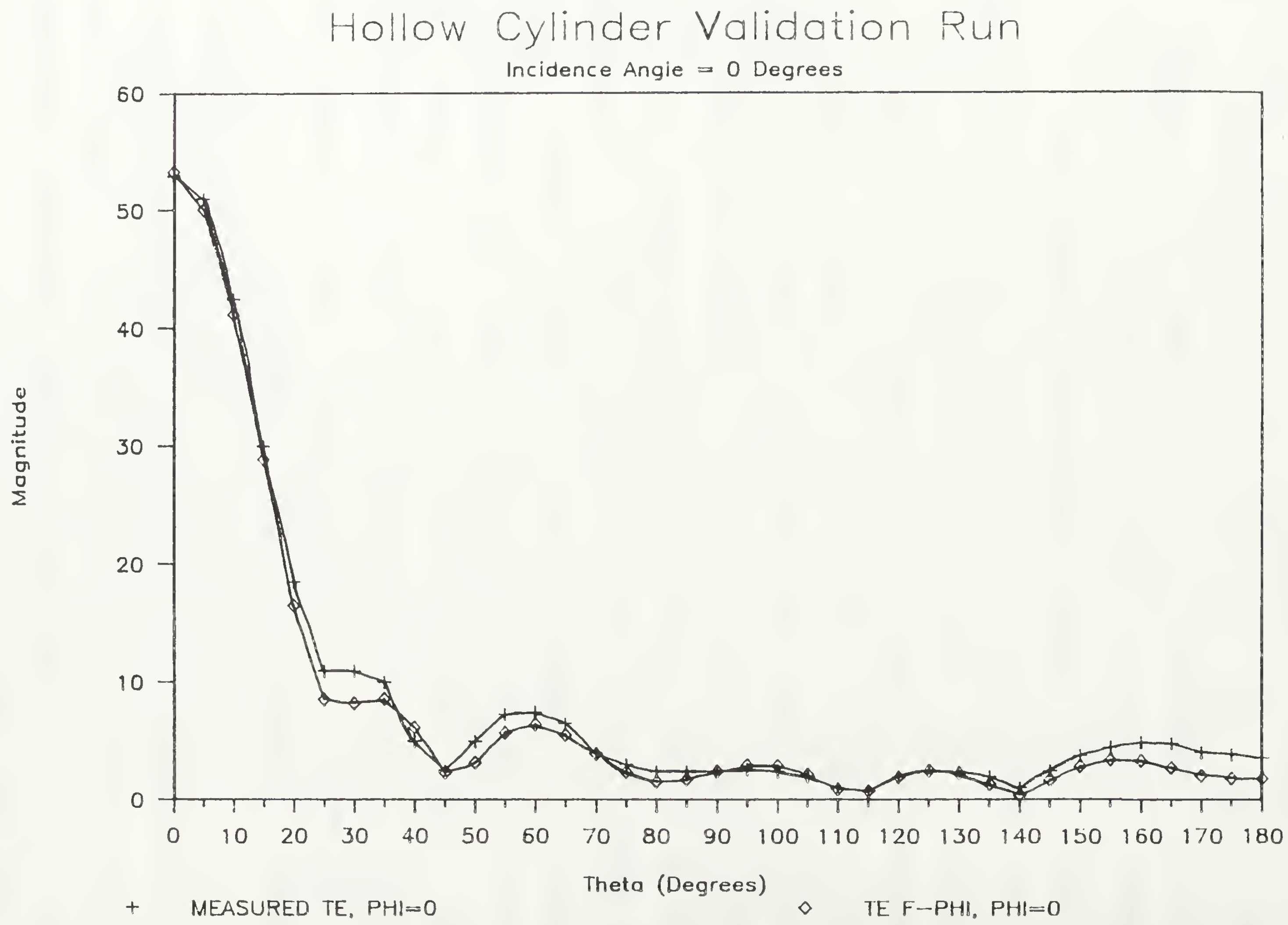


FIG. 5-7. CYLINDER SCATTERING (#1)

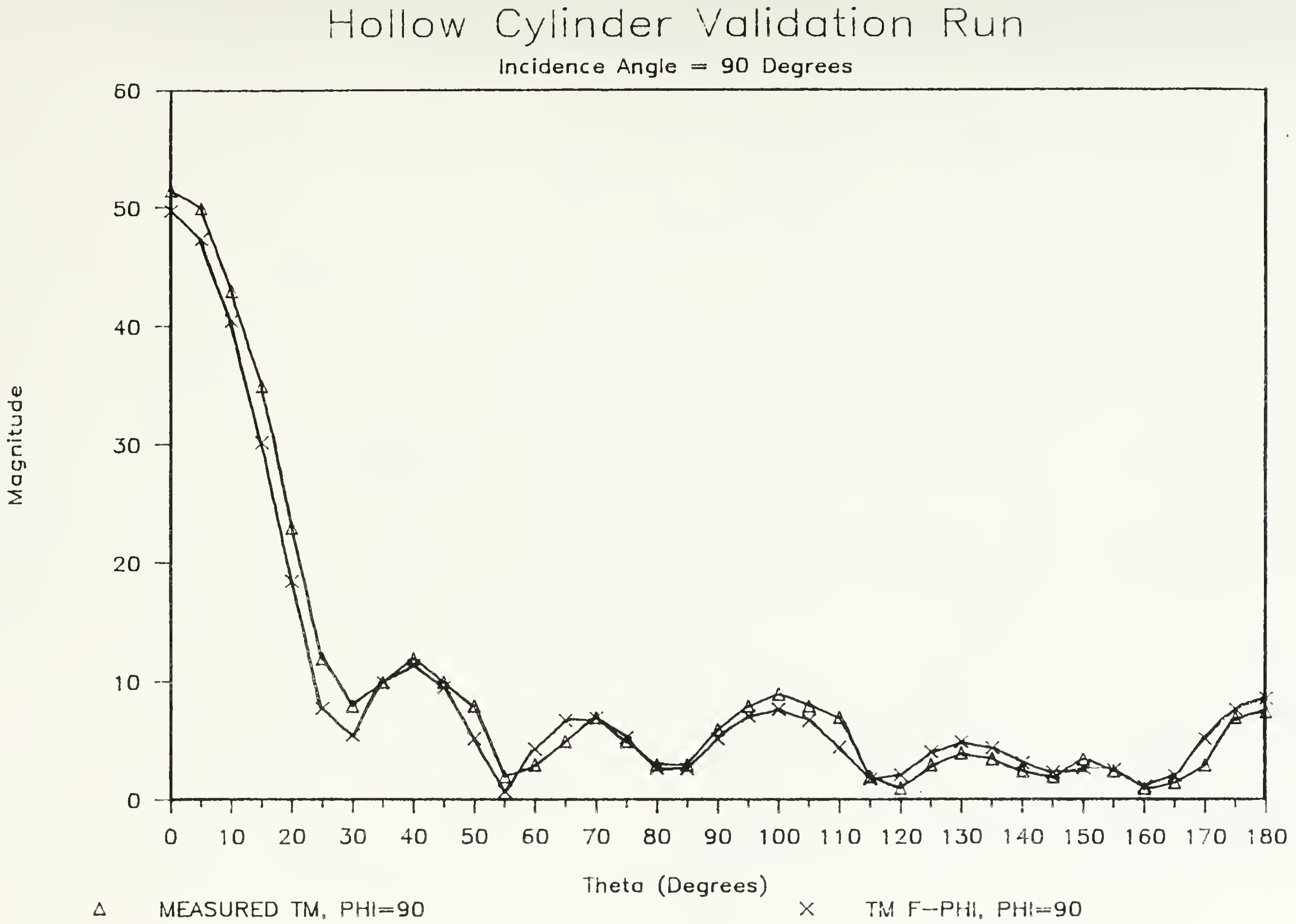


FIG. 5-8. CYLINDER SCATTERING (#2)

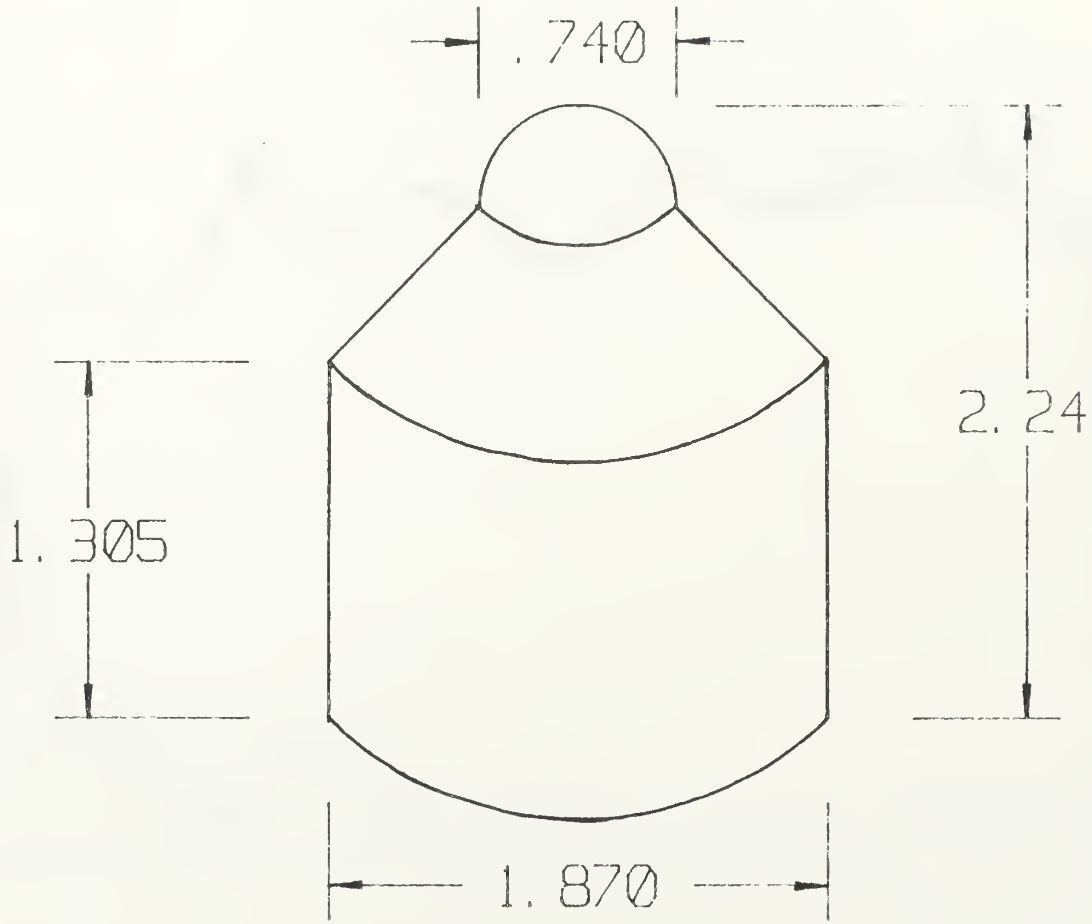


FIG. 5-9. COMPOSITE SCATTERING BODY

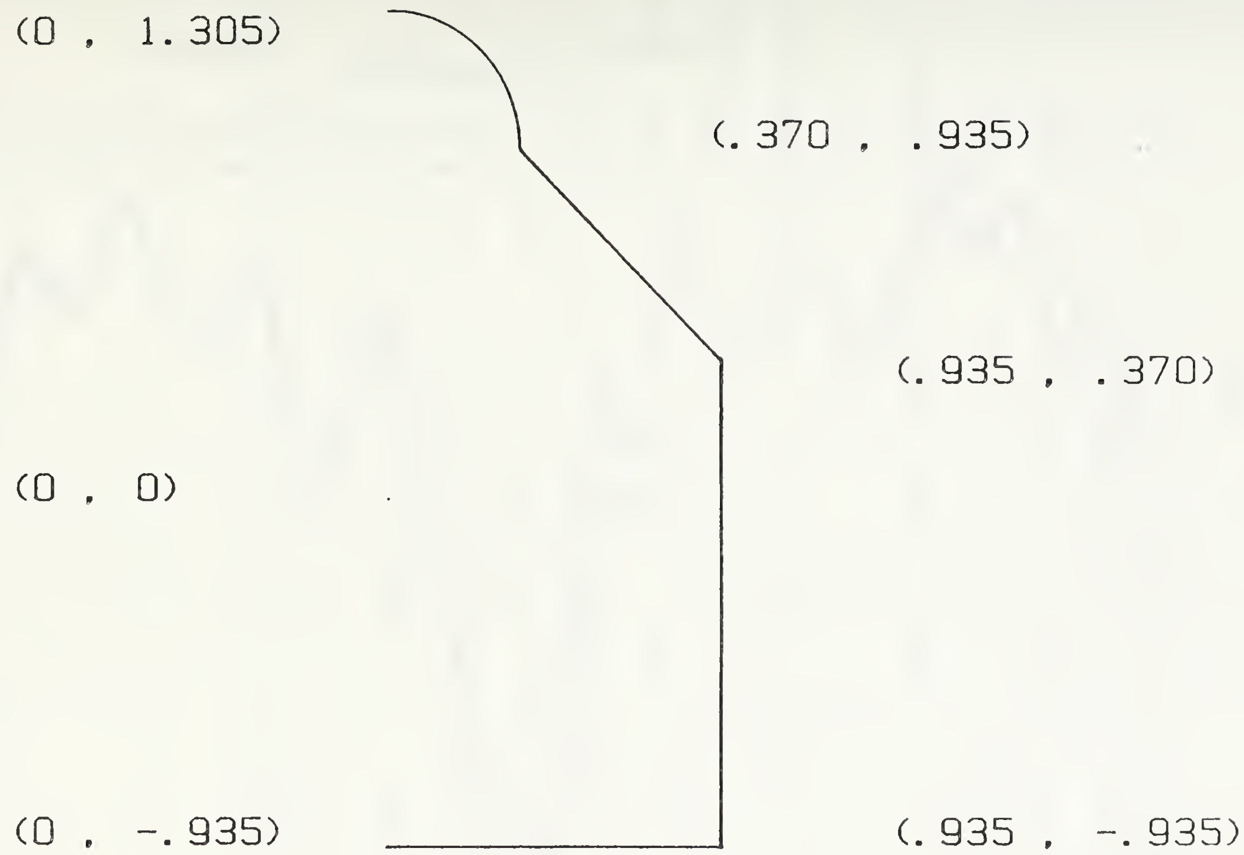


FIG. 5-10. COMPOSITE BODY USED FOR SCATTERING CALCULATION VALIDATION (COORDINATES IN TERMS OF WAVELENGTH)

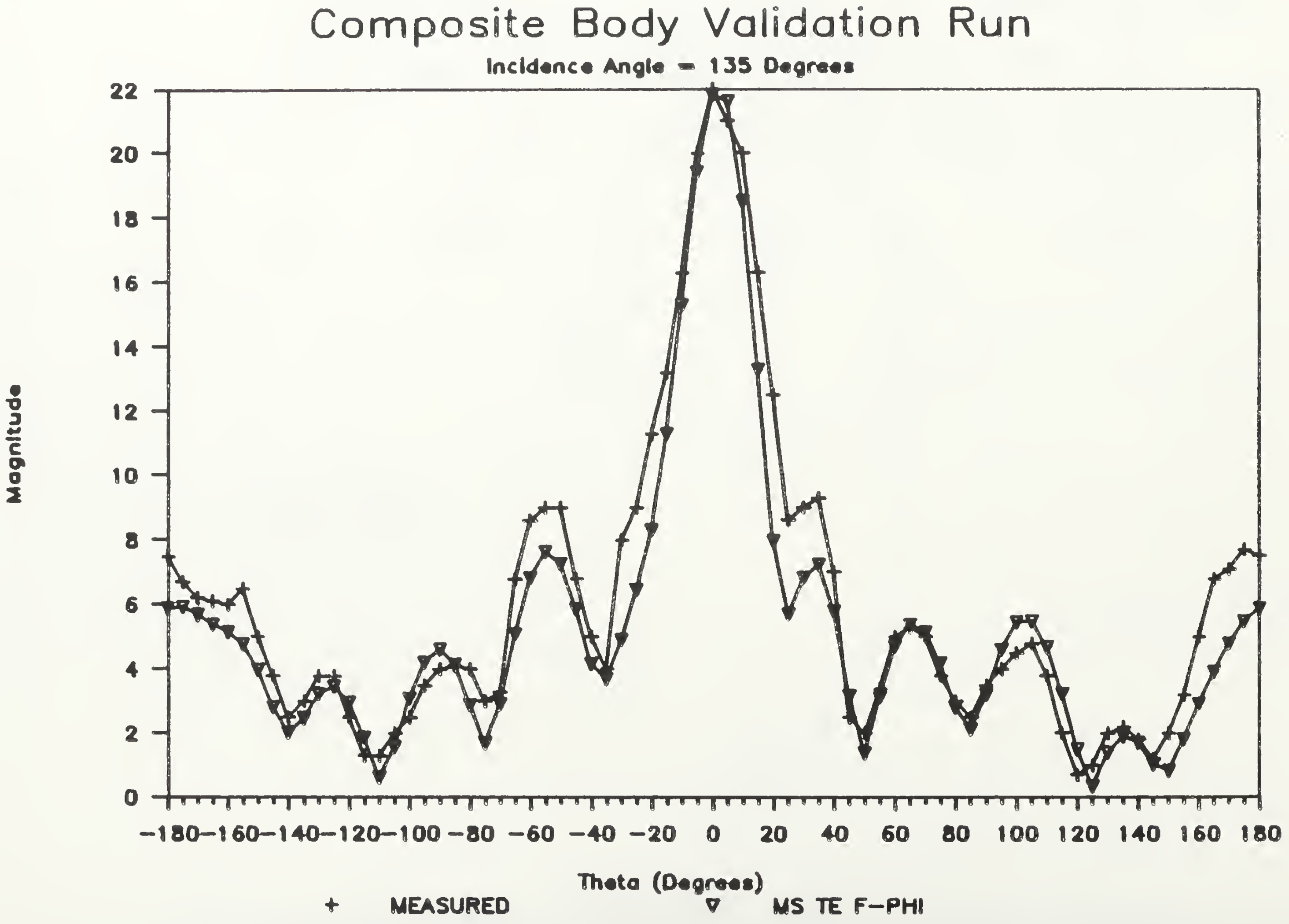


FIG. 5-11. COMPOSITE BODY SCATTERING

In addition to demonstrating the validity of the present version of EMCAD, the composite body of Figure 5-9 helped demonstrate program compatibility. Since the output data of CURVE DIGITIZER was not in the format required for use by EMCAD, the computer program EMCADIN was developed and written to transform the output data of CURVE DIGITIZER into a format which could be used by EMCAD. Figures 5-12, 5-13 and 5-14 present figures which were generated by the programs SPHERE, CONE, and CYLINDER, transformed by CRVDIGIN into proper format, and imported into the computer aided design package CURVE DIGITIZER. The pictures shown are the graphical representations of the imported data as seen by CURVE DIGITIZER. This generation and transformation not only shows the validation of the programs written to generate specific geometric shapes, but also demonstrates the compatibility of these programs with CRVDIGIN, and the compatibility of CRVDIGIN with CURVE DIGITIZER.

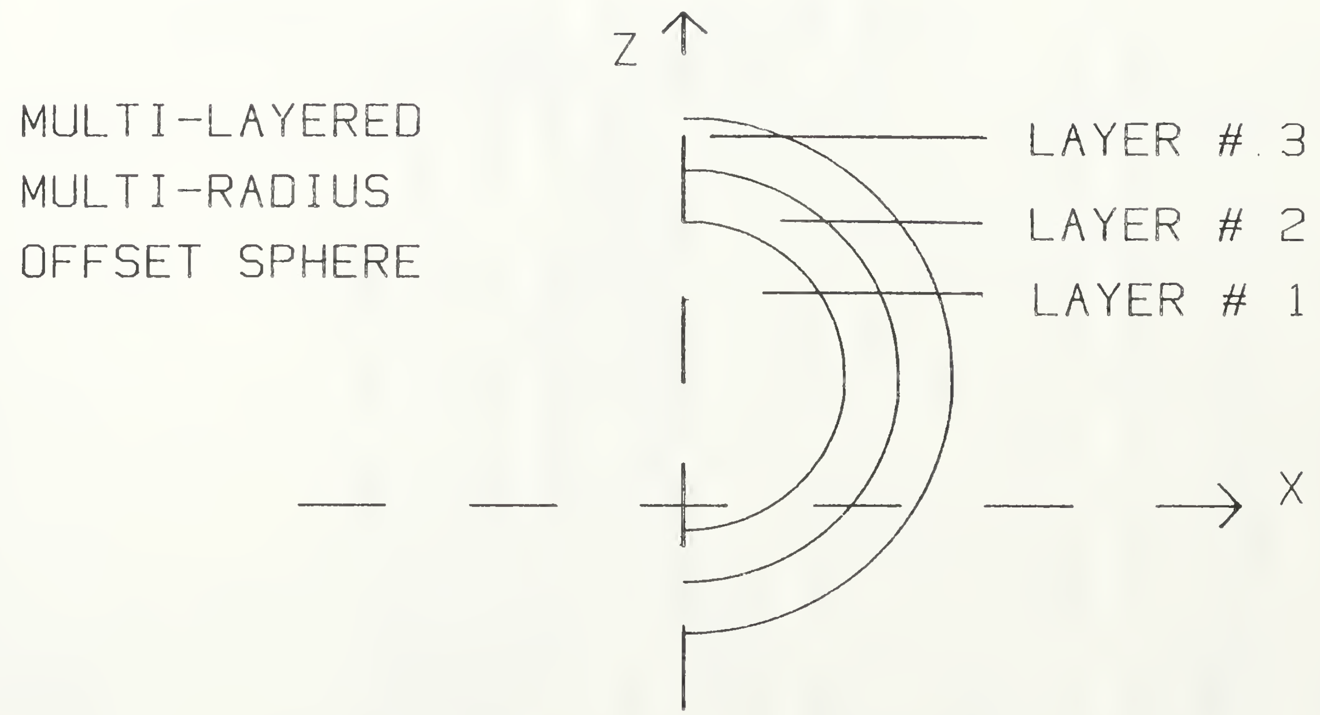


FIG. 5-12. GRAPHICAL REPRESENTATION OF OUTPUT DATA OF PROGRAM SPHERE

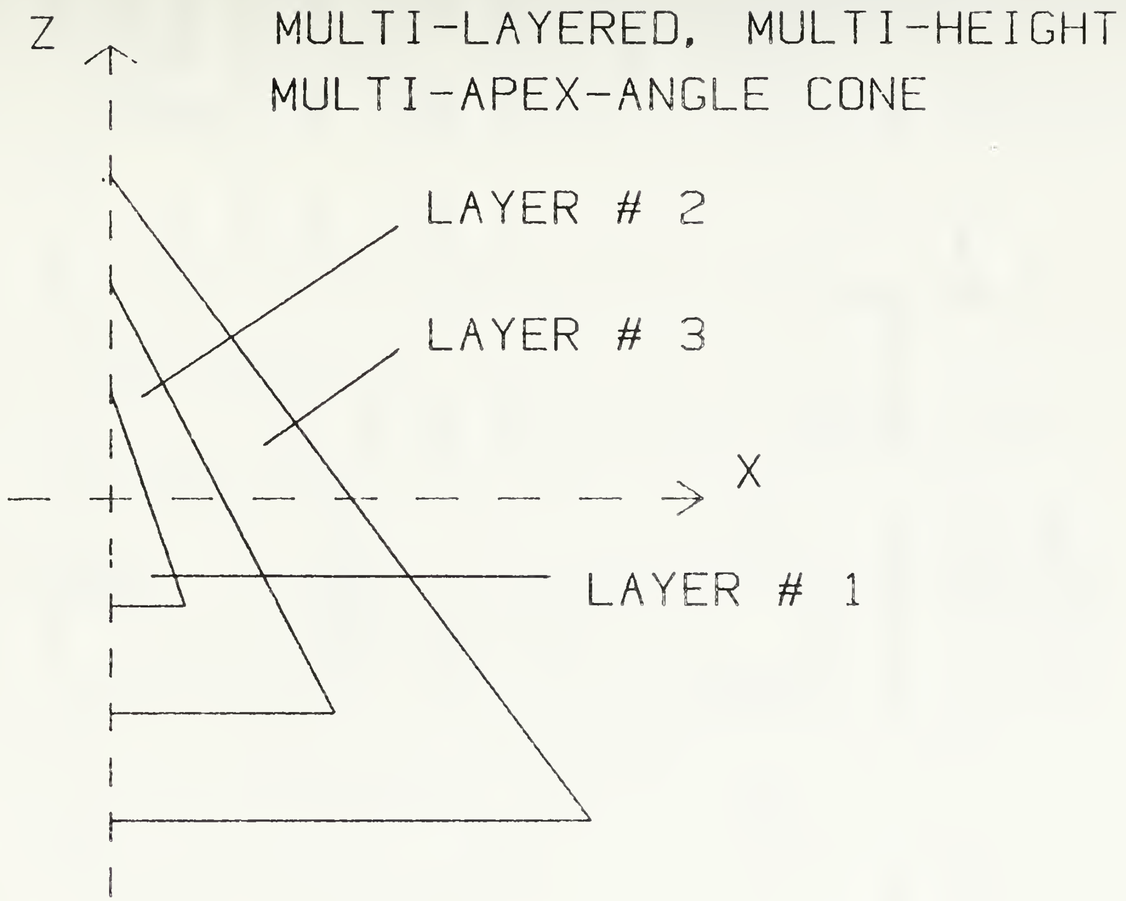


FIG. 5-13. GRAPHICAL REPRESENTATION OF OUTPUT DATA OF PROGRAM CONE

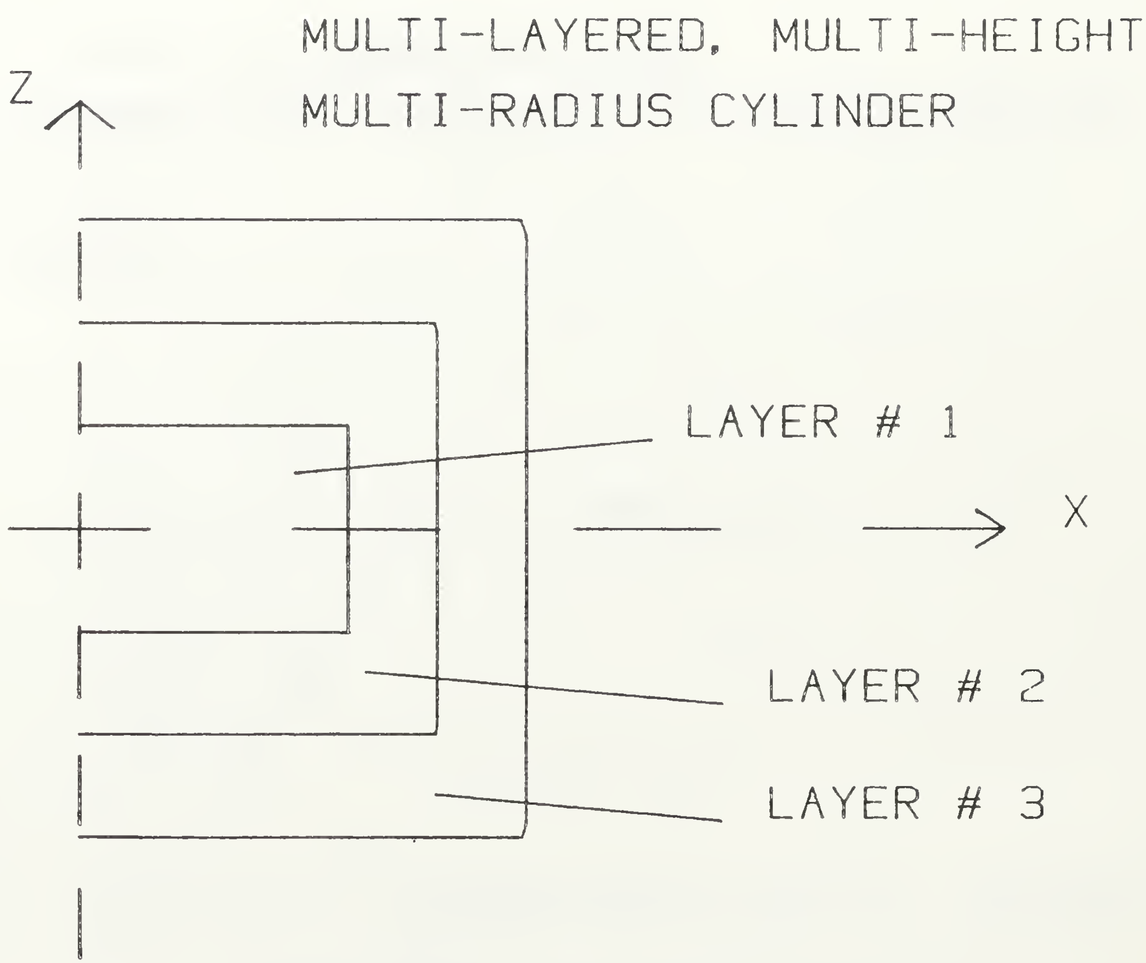


FIG. 5-14. GRAPHICAL REPRESENTATION OF OUTPUT DATA OF PROGRAM CYLINDER

VI. CONCLUSIONS

This thesis has described a very useful numerical algorithm for computing resonance region electromagnetic interaction with arbitrarily shaped and even inhomogeneous axisymmetric penetrable objects. The background material associated with the development of this interactive, finite element scattering algorithm was presented in Chapter II. Included in this background information was the concept of coupled azimuthal potentials, the finite element algorithm and the unimoment method. Chapter III then presented procedures for constructing input data for the scattering program. Computer programs written specifically to generate a given geometry, and to allow data file input and keyboard input were examined in detail. The first half of Chapter IV discussed the extension of the methods used to construct program input data to those methods utilizing graphics. Of special interest was the incorporation of computer aided design features. The second half of the chapter presented user interactive output graphics features. Validations of the modifications and improvements to EMCAD were presented in Chapter V. Extensive graphical comparisons were made between the output of the modified version of EMSCAT, and both theoretical calculations and actual measured experimental data. The impressive accuracy of the

comparisons validated the changes and modifications to EMSCAT used in developing EMCAD. Inter-program compatibility was also extensively demonstrated. The Appendices provide detailed users' manuals for four of the major computer programs developed.

This was only the first stage of a multistage project. Future developments of the axisymmetric code will proceed with major programming modifications to allow metallic surfaces to be embedded in the inhomogeneous penetrable material of the scattering object. Also, the near fields of the object will be made available for display. This is useful for identifying "hot spots" and field caustics.

A second effort has already begun to develop a fully three-dimensional scattering object capability. The basis of this development was the discovery of a generalized coupled azimuthal potential formulation which does not require an axisymmetric material configuration [Ref. 8]. Such a computer algorithm will ultimately employ three-dimensional graphics in a powerful computer aided design package for user design of scattering objects as well as the output of scattering patterns and near field quantities.

APPENDIX A

EMCAD USER'S MANUAL

Welcome to EMCAD. This program computes electromagnetic field scattering from layered bodies of revolution using the coupled azimuthal potential formulation in conjunction with a variational finite element technique and a tri-regional unimoment method. The numerical solution is performed using a two-sweep Riccati transform with circumferential marching.

A. BACKGROUND

EMCAD is based upon an earlier code named EMSCAT which was written during the period 1974-1976 by Dr. M. A. Morgan while at the University of California at Berkeley. Extensive interactive computer aided design (CAD) type features have been added in this effort. These CAD features allow structures composed of multiple dielectric materials to be graphically constructed or modified by the user, and then evaluated by EMCAD for scattering. EMCAD has been run in both Ryan-McFarland FORTRAN (Appendix M) and Microsoft FORTRAN (Appendix M), with the current version using Microsoft. Microcompatibles GRAFMATIC (Appendix M) and later Jewell Technologies GRAFPLUS (Appendix M) were chosen as the commercially available screen graphics and hard copy support for EMCAD output graphics. Because its output

format allowed easy conversion to an acceptable format, West Coast Consultants CURVE DIGITIZER (Appendix M) was chosen as the computer aided design package. Any questions concerning colors, formats, changes to the program, etc... should be referred to the respective user's manuals, Dr. M. A. Morgan or LT. E. M. Connolly (Appendix M).

B. COMPUTER SETUP

EMCAD was written for the following computer configuration:

* IBM AT or compatible, including 80386 based compatibles
* hard disk
* EGA graphics card
* EGA monitor
* math coprocessor - not required, but highly recommended
* memory - 640K RAM minimum

EMCAD can be placed in any directory, at the user's discretion, however, if a RAM-drive is to be used, is highly recommended that during each run the user allocate a minimum of 4M of RAM for EMCAD and EMCAD input/output files. EMCAD input is taken from a variety of sources. The data files from these sources are required to be in the same directory/sub-directory as EMCAD.

C. PROGRAM INPUT

EMCAD requires both keyboard input and data file input. As shown in Figure A-1, two input data files, "MATERIAL".DAT and "STRUCTURE".DAT, are required as input to EMCAD. The quotation marks indicate that the data files can be named at the user's discretion. These two data files contain information which describes the penetrable body of revolution being considered. The "material file" contains the data which conveys the material composition of each layer of the structure. The "structure file" contains the data which defines the structural shape of each layer of the penetrable body. The three methods which are used to generate these data files are discussed in much greater detail in both Chapter II and Chapter III.

D. PROGRAM EXECUTION

EMCAD is a menu driven program. Once the correct directory has been selected, EMCAD can be invoked by typing

```
EMCAD
```

at the DOS prompt. This is shown in the top line of Figure A-2. The program will load itself, and the user will see the next two statements shown in Figure A-2 requesting the names of the "material file" and the "structure file". Sample keyboard inputs are shown for these two requests. It must be pointed out that the extensions are included (.dat)

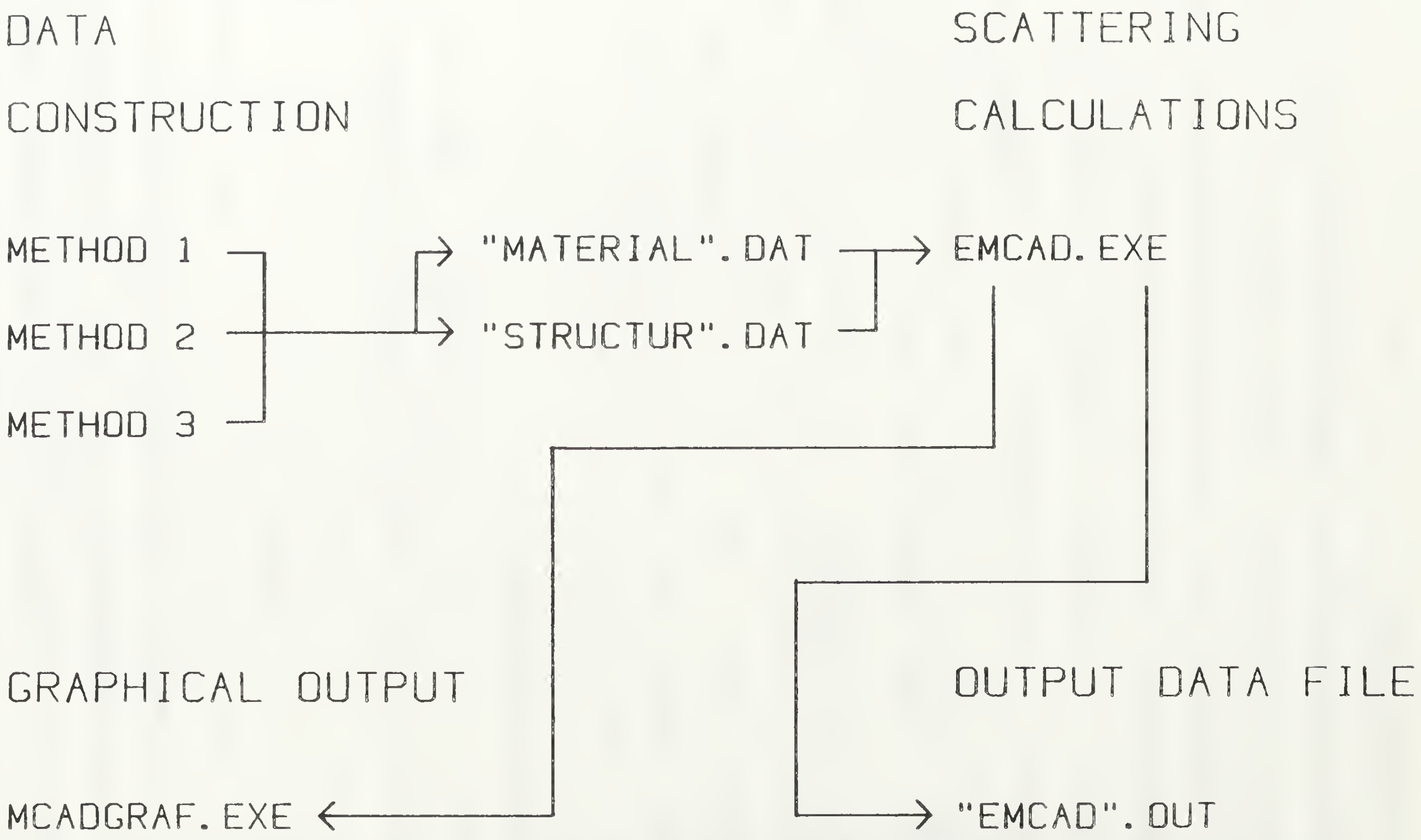


FIG. A-1. SCATTERING CALCULATION INPUT-OUTPUT RELATIONSHIP

```
C:\THESIS\MSFORT>EMCAD


     ENTER MATERIAL PARAMETER FILE (D:FILENAME.EXTENSION) :
mdat1.dat


     ENTER STRUCTURE DATA FILE (D:FILENAME.EXTENSION):
sdat1.dat

Offset Sphere R=0.8 D=0.16
Er=(2.6,0.) Ur=(1.,0.)




     PLEASE PRESS ANY KEY TO CONTINUE.
```

FIG. A-2. INPUT DATA FILES

as required by EMCAD. The two lines beginning with "Offset Sphere..." refer to two label lines which were previously inserted into the "material file" to allow the user to differentiate this set of data from every other set of data. By choosing labels carefully, the user can discern from the very first screen whether the proper set of data is being loaded. This becomes very important because as figures increase in complexity, run time for the program increases. It is not uncommon to have runs which take several hours on an AT class computer. It would be very disturbing to run the program for seven or eight hours only to find out that the incorrect set of data had been used. Upon completion of this screen, the user depresses ENTER to see the following screens:

* Welcome To EMCAD (Figure A-3)

This screen presents some brief background information.

* EMCAD Level of Expertise (Figure A-4)

To facilitate ease in understanding and running the program, EMCAD allows a user to input the necessary information according to the level of expertise needed. For the "Novice" level, the program gives brief explanations of input values, describes the required format, and gives typical values where applicable. The "Expert" level

```
****** WELCOME TO EMCAD ******

THIS PROGRAM COMPUTES EM FIELD SCATTERING FROM PENETRABLE
BODIES OF REVOLUTION USING THE COUPLED AZIMUTHAL POTENTIAL
(CAP) FORMULATION IN CONJUNCTION WITH A VARIATIONAL FINITE-
ELEMENT TECHNIQUE AND A TRI-REGIONAL UNIMOMENT METHOD.  THE
NUMERICAL SOLUTION IS PERFORMED USING A TWO-SWEEP RICCATI
TRANSFORM WITH CIRCUMFERENTIAL MARCHING.
UPDATED VERSION OF ORIGINAL U.C. BERKELEY CODE.
MODIFICATIONS BY M.A. MORGAN    MAR 1987 - FEB 1988
                 E.M. CONNOLLY  JUL 1987 - APR 1988 .



PLEASE PRESS ANY KEY TO CONTINUE.
```

FIG. A-3. WELCOME TO EMCAD

```
******* EMCAD INPUT *******

EMCAD ALLOWS A USER TO INPUT THE NECESSARY INFORMATION
ACCORDING TO THE LEVEL OF HIS EXPERTISE.

1 NOVICE LEVEL - GIVES BRIEF EXPLANATIONS/DESCRIPTIONS
                 OF INPUT VALUES. DESCRIBES THE REQUIRED
                 FORMAT FOR THE INPUT VALUE. GIVES
                 TYPICAL VALUES WHERE APPLICABLE.

2 EXPERT LEVEL - ASSUMES THE USER IS FAMILIAR WITH THE
                 INPUT PARAMETERS AND FORMATS. SIMPLY
                 PROMPTS FOR REQUIRED INPUTS.

3 EXIT EMCAD

PLEASE SELECT THE LEVEL OF EXPERTISE BY ENTERING 1, 2, OR 3.
```

FIG. A-4. EMCAD LEVEL OF USER EXPERTISE

assumes the user is familiar with the input parameters and formats, and simply prompts for the required inputs. The remaining menus will use the "Novice" level to cover all ranges of user.

* Input Name for Output Data File (Figure A-5)

This menu prompts the user for the filename, WITHOUT EXTENSION, that will be used to hold the output data of EMCAD. The program will append the extension .OUT to the user's filename.

* Input Name for Output Graphics Data Files (Figure A-6)

EMCAD produces four output data files for input to the graphing routines of MCADGRAF. All four data files have the same name, but each has a distinct extension related to the type of data the file contains. This menu prompts the user for the name of the output graphics data files, WITHOUT EXTENSION. This filename may be the same as that used for the output data.

* Graphics Caption (Figure A-7)

This menu affords the user the opportunity to input a graphics caption to help identify this set of data from all

THE OUTPUT DATA FILE IS THE FILE CONTAINING THE OUTPUT RESULTS OF ALL NUMERICAL CALCULATIONS CONDUCTED BY EMCAD. THE FORMAT FOR THIS INPUT IS FILENAME ONLY. NO EXTENSION IS REQUIRED OR DESIRED. EMCAD AUTOMATICALLY APPENDS AN EXTENSION OF .OUT TO YOUR FILENAME.

PLEASE ENTER THE FILENAME OF THE OUTPUT DATA FILE.

FIG. A-5. INPUT NAME FOR OUTPUT DATA FILE

THE OUTPUT GRAPHICS-DATA FILE IS THE FILE CONTAINING THE OUTPUT DATA FROM EMCAD TO BE USED AS INPUT TO GRAPHING ROUTINES. THE FORMAT FOR THIS INPUT IS FILENAME ONLY. NO EXTENSION IS REQUIRED OR DESIRED. EMCAD AUTOMATICALLY APPENDS AN EXTENSION OF .TMT, .TMP, .TET, AND .TEP TO YOUR FILENAME AS IT PRODUCES FOUR OUTPUT FILES FOR GRAPHICS.

.TMT -----) TM INCIDENCE, F-THETA
.TMP -----) TM INCIDENCE, F-PHI
.TET -----) TE INCIDENCE, F-THETA
.TET -----) TE INCIDENCE, F-PHI

PLEASE ENTER THE FILE NAME OF THE OUTPUT DATA FILE.

FIG. A-6. INPUT NAME FOR OUTPUT GRAPHICS DATA FILES

THE GRAPHICS CAPTION IS A PERSONALIZED CAPTION ALLOWING THE USER TO IDENTIFY THIS SET OF GRAPHS FROM ALL OTHER SETS. THE MAXIMUM LENGTH OF THIS CAPTION IS 64 CHARACTERS.
NOTE: WHEN USED WITH THE GRAPHICS PACKAGE, THE PROGRAM IS ABLE TO DIFFERENTIATE BETWEEN UPPER CASE AND LOWER CASE CHARACTERS.

PLEASE ENTER ANY GRAPHICS CAPTION YOU DESIRE

FIG. A-7. GRAPHICS CAPTION

other sets. The graphics caption appears on the first line of the header of each graph produced by MCADGRAF.

* Mesh Density (Figure A-8)

As discussed in Chapter II, EMCAD utilizes a semi-annular conformal finite-element mesh in order to conduct scattering calculations. This menu prompts the user for the minimum and maximum mesh density parameters. At azimuthal modes M = 0 and 1, D_{max} is used as the density parameter. The density decreases linearly to D_{min} at the final mode step. Notice that these values are REAL.

* Incidence Angles (Figure A-9)

EMCAD allows up to five incident fields to impinge on the object of interest. This menu asks for the number of incidence angles, and then the value for each in degrees.

* Number of Scattering Field Theta Points (Figure A-10)

EMCAD conducts scattering calculations by starting at $\theta = 0$, and progressing clockwise to $\theta = 180$ in equal increments. This menu allows the user to determine the size of the increments, and indirectly the accuracy.

```
DMIN AND DMAX ARE PARAMETERS OF MESH DENSITY IN TERMS OF
ELEMENTS/INTERIOR LAMBDA. INPUT VALUES ARE EXPECTED TO BE
REAL, I.E. THE DECIMAL POINT MUST BE INCLUDED.
TYPICAL VALUES ARE DMIN = 10. AND DMAX = 15.

PLEASE INPUT DMIN

10.



PLEASE INPUT DMAX

15.
```

FIG. A-8. MESH DENSITY

```
THE NUMBER OF INCIDENT FIELD ANGLES IS THE TOTAL NUMBER OF
INCIDENT FIELDS THAT IMPINGE ON THE OBJECT OF INTEREST.
THIS PROGRAM ALLOWS A MAXIMUM OF FIVE INCIDENT FIELD
ANGLES.  WHEN ENTERING YOUR ANSWER PLEASE DO NOT INCLUDE A
DECIMAL BECAUSE THE INPUT MUST BE IN INTEGER FORMAT, I.E.

         NA = 3  ,  NA = 1

PLEASE INPUT THE NUMBER OF INCIDENT FIELD ANGLES.

2


ENTER INC FLD ANGLE (DEG) FOR #   1

0.


ENTER INC FLD ANGLE (DEG) FOR #   2

90.
```

FIG. A-9. INCIDENCE ANGLES

THE NUMBER OF SCATTERING FIELD THETA POINTS DETERMINES THE SPACING BETWEEN THETA POINTS DURING EMSCAT ITERATIONS AND CALCULATIONS.

DELTA THETA = 180 / (NUMBER THETA POINTS - 1) SO...

NUMBER THETA POINTS = 37 ----> DELTA THETA = 5 DEGREES

NUMBER THETA POINTS = 19 ----> DELTA THETA = 10 DEGREES

WHEN ENTERING YOUR ANSWER, PLEASE DO NOT INCLUDE A DECIMAL BECAUSE THE INPUT MUST BE IN INTEGER FORMAT. I.E. NT = 19

PLEASE INPUT THE NUMBER OF SCATTERING FIELD THETA POINTS

FIG. A-10. NUMBER OF SCATTERING FIELD THETA POINTS

* Phi Angles (Figure A-11)

EMCAD allows up to eight phi angles which define the location from which the user views the scattering for all of the incident angles. This menu asks for the number of phi angles, and then the value for each in degrees.

* Expansion Modes (Figure A-12)

EMCAD partitions the incident and scattered fields into azimuthal modes, indexed by "M". The solutions for these modes are accumulated during the "M-loop". If 0 degrees and 180 degrees are the only incidence angles of interest, then the M = 1 mode is the only mode that is needed. The values of N1 and N2 represent the number of spherical harmonic modes in the spherical core and the external region.

E. PROGRAM OUTPUT

As previously mentioned, EMCAD produces text output data and output data for graphics. The graphics data is discussed in Appendix H, MCADGRAF User's Manual, and the text output data is demonstrated in Appendix B, Portion of an EMCAD Sample Output.

```
THE NUMBER OF PHI ANGLES IS THE TOTAL NUMBER OF PHI ANGLES.
THIS PROGRAM ALLOWS A MAXIMUM OF EIGHT PHI ANGLES.  WHEN
ENTERING YOUR ANSWER PLEASE DO NOT INCLUDE A DECIMAL
BECAUSE THE INPUT MUST BE IN INTEGER FORMAT.  I.E.

    NP = 3  ,  NP = 1

PLEASE INPUT THE NUMBER OF PHI ANGLES.

2


ENTER PHI FLD ANGLE (DEG) FOR #   1

90.


ENTER PHI FLD ANGLE (DEG) FOR #   2

180.
```

FIG. A-11. PHI ANGLES

```
ENTER MSTOP (.LE. 13)

ESTIMATED "MINIMUM" VALUE IS:      1

1


ENTERING NO. OF INTERNAL AND EXTERNAL EXPANSION MODES

ESTIMATED (KI*RMIN) "MINIMUM" INTERNAL N1 IS:    12

ENTER N1  (.GE. MSTOP)

12


ESTIMATED (KO*RMIN) "MINIMUM" INTERNAL N2 IS:    21

ENTER N2  (.GE. MSTOP)

21
```

FIG. A-12. EXPANSION MODES

APPENDIX B

PORTION OF AN EMCAD SAMPLE OUTPUT

This Appendix demonstrates a portion of an EMCAD sample output for the specific case chosen where M indexes from 0 to 7 inclusive. Only the M = 1 case is fully shown, as well as the accumulated solution for all eight modes.

```
******************* EMCAD OUTPUT DATA **************************

       COMPLEX Er(n) AND Ur(n)

           1  (  2.610E+00   -1.000E-06)  (  1.000E+00   -1.000E-06)

       INCIDENT FIELD ANGLES
         N          ALPHA(N)

         1             0.  DEG

         2            90.  DEG

         3           180.  DEG
       SERIES SOLUTION PARAMETERS
       MSTART=    0
       MSTOP=     7
       N1=        8
       N2=       15
```

```
************** M =      1 **************


MESH PARAMETERS
RADII (WAVELENGTHS)
RADMIN=     .64000E+00
RADMAX=     .96000E+00
RADA=       .55241E+00
RADB=       .10476E+01

INTEGER CONSTANTS
NDS=    13
IMX=    59
NR1=     1
NR2=     1

MESH DENSITY AND ELEMENT SIDE RATIOS
DENSITY/LAMBDA-OUT =  24.233
DENSITY/LAMBDA-IN =   15.000
DSA/DR =    .725
DSB/DR =  1.375
```

RICCATI MARCHING STEPS

1	.603E+01
1	.603E+01
2	.602E+01
3	.602E+01
4	.600E+01
5	.599E+01
6	.597E+01
7	.595E+01
8	.592E+01
9	.589E+01
10	.586E+01
11	.583E+01
12	.579E+01
13	.575E+01
14	.571E+01
15	.566E+01
16	.562E+01
17	.557E+01
18	.552E+01
19	.547E+01
20	.542E+01
21	.537E+01
22	.531E+01
23	.526E+01
24	.520E+01
25	.515E+01
26	.509E+01
27	.503E+01
28	.498E+01
29	.492E+01
30	.487E+01
31	.482E+01
32	.477E+01
33	.471E+01
34	.466E+01
35	.461E+01
36	.457E+01
37	.452E+01
38	.448E+01
39	.443E+01
40	.439E+01
41	.435E+01
42	.432E+01
43	.428E+01
44	.425E+01
45	.422E+01
46	.419E+01
47	.416E+01
48	.414E+01
49	.412E+01
50	.410E+01
51	.408E+01
52	.406E+01
53	.405E+01
54	.404E+01
55	.403E+01
56	.403E+01
57	.402E+01
58	.402E+01
59	.402E+01

INTERIOR AND SCATTERED FIELD EXPANSION COEFFICIENTS

AZIMUTHAL MODE NUMBER M= 1

INCIDENCE ANGLE ALPHA = 0. DEGREES

******* INTERIOR COEFFICIENTS *******

STAGGERED COEFFICIENTS: TM INCIDENCE

N	A(N)		B(N)	
1	.500E+00	.266E+00	-.318E+00	.298E+00
2	.209E+00	-.778E-01	.233E+00	.192E+00
3	-.103E+00	-.177E+00	.172E+00	-.660E-01
4	-.207E+00	.582E-01	.163E-01	-.147E+00
5	.996E-02	.169E-01	-.135E+00	.981E-02
6	.886E-01	.390E-01	-.109E+00	.599E-01
7	.614E-01	.138E-01	-.373E-01	.437E-01
8	.197E-01	-.325E-02	-.625E-02	.136E-01

STAGGERED COEFFICIENTS: TE INCIDENCE

N	A(N)		B(N)	
1	.266E+00	-.500E+00	.298E+00	.318E+00
2	-.778E-01	-.209E+00	.192E+00	-.233E+00
3	-.177E+00	.103E+00	-.660E-01	-.172E+00
4	.582E-01	.207E+00	-.147E+00	-.163E-01
5	.169E-01	-.996E-02	.981E-02	.135E+00
6	.390E-01	-.886E-01	.599E-01	.109E+00
7	.138E-01	-.614E-01	.437E-01	.373E-01
8	-.325E-02	-.197E-01	.136E-01	.625E-02

******* SCATTERED COEFFICIENTS *******

STAGGERED COEFFICIENTS: TM INCIDENCE

N	C(N)		D(N)	
1	-.902E-01	-.144E-01	.104E+00	.224E+00
2	-.115E-01	-.195E+00	.955E-02	-.123E+00
3	-.103E+00	.157E+00	-.994E-01	-.850E-01
4	.138E+00	.131E+00	-.928E-01	.696E-01
5	.472E-01	-.954E-01	.670E-01	.158E+00
6	.168E-01	-.616E-01	.510E-01	.659E-01
7	.361E-02	-.171E-01	.159E-01	.135E-01
8	.466E-03	-.306E-02	.298E-02	.160E-02
9	.372E-04	-.416E-03	.398E-03	.103E-03
10	.192E-05	-.497E-04	.451E-04	-.239E-05
11	-.151E-06	-.692E-05	.636E-05	-.951E-06
12	-.606E-07	-.119E-05	.111E-05	-.422E-06
13	-.349E-07	-.207E-06	.255E-06	-.117E-06
14	-.150E-07	-.433E-07	.528E-07	-.342E-07
15	-.336E-08	-.776E-08	.112E-07	-.626E-08

STAGGERED COEFFICIENTS: TE INCIDENCE

N	C(N)		D(N)	
1	-.144E-01	.902E-01	.224E+00	-.104E+00
2	-.195E+00	.115E-01	-.123E+00	-.955E-02
3	.157E+00	.103E+00	-.850E-01	.994E-01
4	.131E+00	-.138E+00	.696E-01	.928E-01
5	-.954E-01	-.472E-01	.158E+00	-.670E-01
6	-.616E-01	-.168E-01	.659E-01	-.510E-01

7	-.171E-01	-.361E-02	.135E-01	-.159E-01
8	-.306E-02	-.466E-03	.160E-02	-.298E-02
9	-.416E-03	-.372E-04	.103E-03	-.398E-03
10	-.497E-04	-.192E-05	-.239E-05	-.451E-04
11	-.692E-05	.151E-06	-.951E-06	-.636E-05
12	-.119E-05	.606E-07	-.422E-06	-.111E-05
13	-.207E-06	.349E-07	-.117E-06	-.255E-06
14	-.433E-07	.150E-07	-.342E-07	-.528E-07
15	-.776E-08	.336E-08	-.626E-08	-.112E-07

INTERIOR AND SCATTERED FIELD EXPANSION COEFFICIENTS

AZIMUTHAL MODE NUMBER M= 1

INCIDENCE ANGLE ALPHA = 90. DEGREES

******* INTERIOR COEFFICIENTS *******

STAGGERED COEFFICIENTS: TM INCIDENCE

N	A(N)		B(N)	
1	.457E+00	.212E+00	.555E-02	.830E-01
2	.913E-02	-.474E-01	-.246E+00	-.152E+00
3	.354E-01	.375E-01	-.262E-01	.429E-01
4	.260E-03	-.239E-02	-.588E-01	-.703E-01
5	.650E-02	.110E-01	-.653E-01	-.473E-01
6	.782E-02	.751E-02	.204E-02	-.190E-01
7	.198E-02	.290E-02	.651E-02	-.626E-02
8	.708E-03	.104E-02	.371E-02	-.277E-02

STAGGERED COEFFICIENTS: TE INCIDENCE

N	A(N)		B(N)	
1	.662E-01	-.344E-01	.624E+00	.267E-01
2	.294E+00	.353E-01	.397E-01	-.379E-01
3	.833E-01	-.504E-01	.342E-01	.258E-01
4	.445E-01	.346E-01	.219E-01	-.115E-01
5	.653E-01	.237E-01	-.173E-02	.119E-01
6	-.126E-01	.127E-01	-.561E-03	.916E-02
7	-.131E-01	.504E-02	-.896E-03	.338E-02
8	-.574E-02	.199E-02	-.384E-03	.100E-02

******* SCATTERED COEFFICIENTS *******

STAGGERED COEFFICIENTS: TM INCIDENCE

N	C(N)		D(N)	
1	.901E-01	-.229E+00	.536E-01	-.517E-01
2	-.376E-01	.200E-01	-.105E+00	.169E+00
3	.411E-01	-.318E-01	.400E-01	-.698E-02
4	-.387E-03	-.606E-03	-.847E-01	.700E-01
5	.127E-01	-.868E-02	-.342E-01	.348E-01
6	.497E-02	-.408E-02	-.895E-02	-.529E-02
7	.103E-02	-.806E-03	-.195E-02	-.393E-02
8	.164E-03	-.112E-03	-.297E-03	-.105E-02
9	.124E-04	-.197E-05	-.402E-04	-.181E-03
10	.339E-05	.149E-06	-.346E-05	-.265E-04
11	-.352E-06	.528E-06	-.403E-06	-.292E-05
12	.186E-06	.523E-07	-.977E-07	-.256E-06
13	-.314E-07	.132E-07	.357E-08	-.189E-07
14	.795E-08	.301E-08	-.602E-08	-.652E-09
15	-.137E-08	-.282E-09	.730E-09	-.169E-09

STAGGERED COEFFICIENTS: TE INCIDENCE

N	C(N)		D(N)	
1	-.758E-02	.291E-01	.722E-02	.198E-03
2	.260E-01	-.267E-01	-.152E-01	.294E-01
3	-.582E-01	.473E-01	.212E-01	-.237E-01
4	.726E-01	-.648E-01	-.108E-01	-.879E-02
5	.301E-01	-.344E-01	.157E-01	.513E-02
6	.768E-02	.348E-02	.667E-02	.194E-02

7	.162E-02	.324E-02	.143E-02	.572E-03
8	.236E-03	.848E-03	.222E-03	.114E-03
9	.330E-04	.135E-03	.187E-04	.261E-04
10	.241E-05	.186E-04	.394E-05	.294E-05
11	.301E-06	.171E-05	-.363E-06	.832E-06
12	.583E-07	.955E-07	.176E-06	.708E-07
13	-.925E-08	-.222E-08	-.374E-07	.113E-07
14	.457E-08	-.206E-08	.757E-08	.176E-08
15	-.860E-09	-.157E-09	-.154E-08	-.528E-09

ACCUMULATED FORWARD AND BACK SCATTERED AMPLITUDES

M = 1

STAGGERED AMPS: INCIDENT TM

ALPHA (DEG)	FWD		BK	
0.	-.805E+01	-.177E+02	-.139E+01	-.550E+01
90.	-.294E+01	-.359E+01	.200E+01	.141E+01
180.	-.776E+01	-.172E+02	-.371E+01	.474E+01

STAGGERED AMPS: INCIDENT TE

ALPHA (DEG)	FWD		BK	
0.	-.805E+01	-.177E+02	-.139E+01	-.550E+01
90.	-.213E+01	-.295E+01	.196E-01	-.247E+00
180.	-.776E+01	-.172E+02	-.371E+01	.474E+01

SCATTERING CROSS SECTION CONVERGENCE

M= 1

NINC = 1

N	NEW TERM	PCT CHANGE
1	.6E-01	100.0000
2	.2E+00	80.6270
3	.7E+00	69.3592
4	.1E+01	58.7450
5	.2E+01	47.0760
6	.1E+01	17.3836
7	.1E+00	1.7682
8	.4E-02	.0730
9	.9E-04	.0017
10	.2E-05	.0000
11	.4E-07	.0000
12	.2E-08	.0000
13	.1E-09	.0000
14	.6E-11	.0000
15	.3E-12	.0000

NINC = 2

N	NEW TERM	PCT CHANGE
1	.6E-01	100.0000
2	.2E+00	77.2227
3	.6E-01	18.8104
4	.3E+00	52.9934
5	.1E+00	17.4571
6	.1E-01	1.6261
7	.3E-02	.3495
8	.2E-03	.0301
9	.9E-05	.0012
10	.3E-06	.0000
11	.4E-08	.0000
12	.7E-10	.0000
13	.1E-11	.0000
14	.1E-12	.0000
15	.3E-14	.0000

NINC = 3

N	NEW TERM	PCT CHANGE
1	.5E-01	100.0000
2	.3E+00	85.8396
3	.1E+01	76.4532
4	.3E+01	63.6074
5	.1E+01	22.9305
6	.2E+00	4.1769
7	.1E-01	.1771
8	.3E-03	.0063
9	.1E-04	.0002
10	.4E-06	.0000
11	.9E-08	.0000
12	.4E-09	.0000
13	.2E-10	.0000
14	.1E-11	.0000
15	.5E-13	.0000

NINC = 4

N	NEW TERM	PCT CHANGE
1	.6E-01	100.0000
2	.2E+00	80.6270
3	.7E+00	69.3592
4	.1E+01	58.7450
5	.2E+01	47.0760
6	.1E+01	17.3836
7	.1E+00	1.7682
8	.4E-02	.0730
9	.9E-04	.0017
10	.2E-05	.0000
11	.4E-07	.0000
12	.2E-08	.0000
13	.1E-09	.0000
14	.6E-11	.0000
15	.3E-12	.0000

NINC = 5

N	NEW TERM	PCT CHANGE
1	.8E-03	100.0000
2	.1E-01	93.3376
3	.9E-01	87.6869
4	.3E+00	73.3983
5	.1E+00	24.8194
6	.1E-01	2.0363
7	.2E-02	.4062
8	.2E-03	.0320
9	.6E-05	.0011
10	.1E-06	.0000
11	.2E-08	.0000
12	.3E-10	.0000
13	.1E-11	.0000
14	.8E-13	.0000
15	.4E-14	.0000

MODAL CROSS SECTION EVOLUTION

NORMALIZED W.R.T. WAVELENGTH SQUARED

INCIDENCE ANGLE = 0. DEGREES

	INCIDENT TM POLARIZATION			INCIDENT TE POLARIZATION		
M	SCT	ABS	TOT	SCT	ABS	TOT
0	.000E+00	.000E+00	.000E+00	.000E+00	.000E+00	.000E+00
1	.558E+01	.705E-01	.565E+01	.558E+01	.705E-01	.565E+01
2	.558E+01	.705E-01	.565E+01	.558E+01	.705E-01	.565E+01
3	.558E+01	.705E-01	.565E+01	.558E+01	.705E-01	.565E+01
4	.558E+01	.705E-01	.565E+01	.558E+01	.705E-01	.565E+01
5	.558E+01	.705E-01	.565E+01	.558E+01	.705E-01	.565E+01
6	.558E+01	.705E-01	.565E+01	.558E+01	.705E-01	.565E+01
7	.558E+01	.705E-01	.565E+01	.558E+01	.705E-01	.565E+01

INCIDENCE ANGLE = 90. DEGREES

	INCIDENT TM POLARIZATION			INCIDENT TE POLARIZATION		
M	SCT	ABS	TOT	SCT	ABS	TOT
0	.247E+00	.100E+00	.347E+00	.326E+00	.105E+00	.431E+00
1	.104E+01	.998E-01	.114E+01	.834E+00	.105E+00	.940E+00
2	.197E+01	.110E+00	.208E+01	.202E+01	.110E+00	.213E+01
3	.317E+01	.106E+00	.328E+01	.285E+01	.106E+00	.296E+01
4	.440E+01	.113E+00	.451E+01	.396E+01	.118E+00	.407E+01
5	.521E+01	.144E+00	.536E+01	.519E+01	.199E+00	.539E+01
6	.525E+01	.166E+00	.542E+01	.523E+01	.216E+00	.545E+01
7	.525E+01	.166E+00	.542E+01	.523E+01	.216E+00	.545E+01

INCIDENCE ANGLE = 180. DEGREES

	INCIDENT TM POLARIZATION			INCIDENT TE POLARIZATION		
M	SCT	ABS	TOT	SCT	ABS	TOT
0	.220E-12	-.220E-12	.000E+00	.209E-12	-.209E-12	.000E+0
1	.546E+01	.255E-01	.548E+01	.546E+01	.255E-01	.548E+0
2	.546E+01	.255E-01	.548E+01	.546E+01	.255E-01	.548E+0
3	.546E+01	.255E-01	.548E+01	.546E+01	.255E-01	.548E+0
4	.546E+01	.255E-01	.548E+01	.546E+01	.255E-01	.548E+0
5	.546E+01	.255E-01	.548E+01	.546E+01	.255E-01	.548E+0
6	.546E+01	.255E-01	.548E+01	.546E+01	.255E-01	.548E+0
7	.546E+01	.255E-01	.548E+01	.546E+01	.255E-01	.548E+0

***** SCATTERED FIELD COMPLEX AMPLITUDES *****

INCIDENCE ANGLE = 0. DEGREES

PHI= 90. DEGREES

STAGGERED RAD FLDS: TM INCIDENCE

THETA	F-THETA MAG	F-THETA DEG	F-PHI MAG	F-PHI DEG
0.	.851E-06	.00	.195E+02	65.60
5.	.821E-06	.00	.187E+02	65.19
10.	.735E-06	.00	.166E+02	63.93
15.	.608E-06	.00	.133E+02	61.71
20.	.461E-06	.00	.925E+01	58.17
25.	.322E-06	.00	.499E+01	51.69
30.	.229E-06	.00	.114E+01	19.66
35.	.207E-06	.00	.253E+01	-112.81
40.	.225E-06	.00	.484E+01	-124.99
45.	.239E-06	.00	.603E+01	-131.99
50.	.235E-06	.00	.608E+01	-138.30
55.	.217E-06	.00	.515E+01	-144.73
60.	.197E-06	.00	.353E+01	-151.82
65.	.183E-06	.00	.159E+01	-161.86
70.	.173E-06	.00	.382E+00	49.46
75.	.160E-06	.00	.187E+01	18.57
80.	.138E-06	.00	.281E+01	10.27
85.	.111E-06	.00	.307E+01	3.63
90.	.933E-07	.00	.270E+01	-2.13
95.	.993E-07	.00	.186E+01	-6.50
100.	.120E-06	.00	.812E+00	-5.21
105.	.137E-06	.00	.290E+00	118.65
110.	.138E-06	.00	.996E+00	140.86
115.	.121E-06	.00	.133E+01	138.49
120.	.889E-07	.00	.121E+01	133.24
125.	.557E-07	.00	.716E+00	123.28
130.	.583E-07	.00	.189E+00	39.52
135.	.939E-07	.00	.754E+00	-30.93
140.	.129E-06	.00	.125E+01	-35.32
145.	.153E-06	.00	.142E+01	-29.65
150.	.165E-06	.00	.134E+01	-9.95
155.	.173E-06	.00	.149E+01	26.89
160.	.184E-06	.00	.231E+01	55.01
165.	.202E-06	.00	.348E+01	67.53
170.	.224E-06	.00	.460E+01	72.91
175.	.241E-06	.00	.539E+01	75.20
180.	.248E-06	.00	.568E+01	75.85

***** SCATTERED FIELD COMPLEX AMPLITUDES *****

INCIDENCE ANGLE = 0. DEGREES

PHI= 90. DEGREES

STAGGERED RAD FLDS: TE INCIDENCE

THETA	F-THETA		F-PHI	
	MAG	DEG	MAG	DEG
0.	.195E+02	-114.40	.851E-06	.00
5.	.188E+02	-115.57	.819E-06	.00
10.	.168E+02	-119.26	.724E-06	.00
15.	.139E+02	-126.13	.580E-06	.00
20.	.105E+02	-137.69	.404E-06	.00
25.	.738E+01	-157.34	.218E-06	.00
30.	.524E+01	169.80	.497E-07	.00
35.	.473E+01	128.60	.110E-06	.00
40.	.515E+01	95.19	.212E-06	.00
45.	.547E+01	69.76	.264E-06	.00
50.	.537E+01	46.52	.266E-06	.00
55.	.497E+01	21.84	.225E-06	.00
60.	.451E+01	-5.65	.154E-06	.00
65.	.419E+01	-34.83	.693E-07	.00
70.	.396E+01	-63.23	.167E-07	.00
75.	.366E+01	-90.06	.818E-07	.00
80.	.316E+01	-117.25	.123E-06	.00
85.	.255E+01	-149.31	.134E-06	.00
90.	.214E+01	168.63	.118E-06	.00
95.	.227E+01	123.71	.814E-07	.00
100.	.275E+01	89.95	.355E-07	.00
105.	.314E+01	65.78	.127E-07	.00
110.	.317E+01	45.98	.435E-07	.00
115.	.277E+01	26.74	.579E-07	.00
120.	.203E+01	3.03	.527E-07	.00
125.	.127E+01	-39.54	.313E-07	.00
130.	.133E+01	-107.69	.828E-08	.00
135.	.215E+01	-147.00	.329E-07	.00
140.	.295E+01	-169.25	.545E-07	.00
145.	.349E+01	172.65	.619E-07	.00
150.	.378E+01	154.61	.585E-07	.00
155.	.396E+01	135.25	.653E-07	.00
160.	.421E+01	115.51	.101E-06	.00
165.	.462E+01	98.19	.152E-06	.00
170.	.512E+01	85.58	.201E-06	.00
175.	.552E+01	78.23	.236E-06	.00
180.	.568E+01	75.85	.248E-06	.00

APPENDIX C

EMCAD SOURCE CODE

```
      PROGRAM EMCAD
C
C     ******************************************************************
C
C     THIS PROGRAM COMPUTES EM FIELD SCATTERING FROM LAYERED
C     BODIES OF REVOLUTION USING THE COUPLED AZIMUTHAL POTENTIAL
C     (CAP) FORMULATION IN CONJUNCTION WITH A VARIATIONAL FINITE-
C     ELEMENT TECHNIQUE AND A TRI-REGIONAL UNIMOMENT METHOD.  THE
C     NUMERICAL SOLUTION IS PERFORMED USING A TWO-SWEEP RICCATI
C     TRANSFORM WITH CIRCUMFERENTIAL MARCHING.   UNIT 1 IS USED FOR
C     SEQUENTIAL UNFORMATTED STORAGE OF GENERATED RICCATI ARRAYS TO
C     DISK.  UPDATED VERSION OF CDC-7600 U.C. BERKELEY CODE "EMSCAT"
C     WITH LAYERED MEDIA ADDITIONS.
C
C                 MODS BY M.A. MORGAN    MAR 87-FEB 88
C                         E.M. CONNOLLY  MAR 87-FEB 88.
C
C       ****************** EMCAD PARAMETERS ***********************
C
C       **   NA - NO. OF INCIDENT ANGLES, DALPHA(J) (DEGREES)
C       **   DMIN,DMAX - MIN AND MAX MESH DENSITY (ELEM/LAMBDA)
C       **   NR1,NR2 - RADIAL NODE SPACING BETWEEN RA-R1 & R2-RB
C       **   NLAY - NO. OF LAYERED SURFACES,  NLAY+1 IS FREE SPACE
C       **   ER(NLAY),UR(NLAY) - COMPLEX ER AND UR OF PENETRABLE LAYERS
C       **   MSTART,MSTOP - AZIMUTHAL MODE RANGE ON "M"
C       **   N1,N2 - UPPER LIMITS ON INTERIOR & EXTERIOR MODES "N"
C       **   NT,NP - NO. OF FIELD PRINTOUT THETA & PHI POINTS
C       **   NB,NBMX - NO. OF INTERIOR NODAL POTENTIALS/THETA STEP
C       **   NCMAX - NO. OF INTERIOR + EXTERIOR EXPANSION COEF'S
C       **   NMMAX - NCMAX + 2*NA (NO. UNIMOMENT BC'S)
C       **   NINC,NINMX - 2*NA (2 POLARIZATIONS/INC ANGLE)
C       **   BC - 0 OR 1 : Z-AXIS HOMO DIRICHLET OR CAUCHY BC'S
C       **   IMX,IBIG - NO. OF THETA STEPS IN MARCHING & DATA FILE
C
C       ****************** ARRAY DIMENSIONS ************************
C
C       **   R,RINV - (NBMX,NBMX)   RICCATI MATICES
C       **   S,SINV - (NBMX,NMMAX)  S-VECTOR ARRAYS
C       **   FMOM,DUM - (NCMAX,NCMAX)   COEFFICIENT MOMENT ARRAYS
C       **   DMOM,COEF - (NCMAX,NINMX)   DRIVING & COEF ARRAYS (SAME)
C       **   FT1 TO FP2 - (NT,NP,NA)   SCATTERED E-FIELDS
C       **   RAD - (2,5)   RADII(THETA) OF LAYERS AT I & I+1 STEPS
C
C       ****************************************************************
C
      CHARACTER*1 BELL,FF,LEVEL
      CHARACTER*8 GRFDAT,PRTDAT
      CHARACTER*12 GRFDT1,GRFDT2,GRFDT3,GRFDT4,PRTDT1
      CHARACTER*64 DUMMY,GRAFLAB,STRDAT,HDR,MATPAR
      COMPLEX A,B,G,P,E1,H1,E2,H2,EB1,HB1,EB2,HB2
      COMPLEX F30,F21,F12,F20,F11,F10,F1,F2,FT1,FP1,FT2,FP2
      COMPLEX R(58,58),S(58,80),RINV(58,58),SINV(58,80),ER(6),UR(6)
      COMPLEX FMOM(70,70),COEF(70,10)
      INTEGER BC,ELND,IDUMMY
      REAL RAD(2,5)
      COMMON/ONE/X(62),Y(62)
      COMMON/TWO/ELND(60,3),NDEL(62,3),NDCT(62)
      COMMON/THREE/EB1(3,80),HB1(3,80),EB2(3,80),HB2(3,80)
```

```
      COMMON/FOUR/A(58,4),B(58,6),G(58,4),P(4,80),KA(58),KB(58),KG(58)
      COMMON/FIVE/E1(60,3,3),H1(60,3,3),E2(60,3,3),H2(60,3,3)
      COMMON/SIX/FT1(37,8,5),FP1(37,8,5),FT2(37,8,5),FP2(37,8,5)
      COMMON/SEVEN/SCAT(13,10),ABSB(13,10),TOTL(13,10)
      COMMON/EIGHT/ALPHA(5),PHI(8),DALPHA(5),DPHI(8)
      COMMON/TEN/F30,F21,F12,F20,F11,F10,P30,P21,P12,P20,P11,P10
      EQUIVALENCE (RINV,SINV)
      BELL=CHAR(7)
      FF = CHAR(12)
C     SETTING MAXIMUM MATRIX DIMENSIONS
      NCMAX=70
      NMMAX=80
      NBMX=58
      NINMX=10
      PI=3.1415927
      DTR=PI/180.
C     READING INC. FIELD, MESH, AND SOLUTION PARAMETERS
      WRITE(*,137)
      READ(*,101) MATPAR
      OPEN(2,FILE=MATPAR)
      WRITE(*,112)
      READ(*,101) STRDAT
      OPEN(10,FILE=STRDAT)
      READ(2,101) HDR
      WRITE(*,101)
      WRITE(*,101) HDR
      READ(2,101) HDR
      WRITE(*,101) HDR
      WRITE(*,25)
      READ(*,101) DUMMY
      WRITE(*,101) FF
      WRITE(*,10)
      READ(*,101) DUMMY
      WRITE(*,101) FF
      WRITE(*,12)
      READ(*,101) LEVEL
      WRITE(*,101) FF
      IF(LEVEL.EQ.'3') GO TO 99
      IF(LEVEL.EQ.'2') THEN
         WRITE(*,113)
         READ(*,101) PRTDAT
      ELSE
         WRITE(*,13)
         READ(*,101) PRTDAT
      ENDIF
      PRTDT1 = PRTDAT//'.OUT'
      OPEN(3,FILE=PRTDT1)
      IF(LEVEL.EQ.'2') THEN
         WRITE(*,114)
         READ(*,101) GRFDAT
      ELSE
         WRITE(*,14)
         READ(*,101) GRFDAT
      ENDIF
      GRFDT1 = GRFDAT//'.TMT'
      GRFDT2 = GRFDAT//'.TMP'
      GRFDT3 = GRFDAT//'.TET'
      GRFDT4 = GRFDAT//'.TEP'
      OPEN(4,FILE=GRFDT1)
      OPEN(7,FILE=GRFDT2)
      OPEN(8,FILE=GRFDT3)
      OPEN(9,FILE=GRFDT4)
      IF(LEVEL.EQ.'2') THEN
         WRITE(*,115)
         READ(*,101) GRAFLAB
      ELSE
```

```
         WRITE(*,15)
         READ(*,101) GRAFLAB
      ENDIF
      WRITE(4,101) GRAFLAB
      WRITE(7,101) GRAFLAB
      WRITE(8,101) GRAFLAB
      WRITE(9,101) GRAFLAB
      IF(LEVEL.EQ.'2') THEN
         WRITE(*,116)
         READ(*,*) DMIN
         WRITE(*,117)
         READ(*,*) DMAX
      ELSE
         WRITE(*,16)
         READ(*,*) DMIN
         WRITE(*,17)
         READ(*,*) DMAX
      ENDIF
      WRITE(3,101) FF
      WRITE(3,100)
      WRITE(3,*)
      WRITE(3,111)
      READ(2,102) NLAY
      DO 22 L=1,NLAY
      READ(2,103) ER(L),UR(L)
      WRITE(3,107) L,ER(L),UR(L)
22    CONTINUE
      READ(10,138) IBIG
      READ(10,103) RMIN,RMAX
      IF(LEVEL.EQ.'2') THEN
         WRITE(*,118)
         READ(*,*) NA
      ELSE
         WRITE(*,18)
         READ(*,*) NA
      ENDIF
      NINC=2*NA
      SAMX=0.0
      WRITE(3,109)
      DO 33 I=1,NA
      WRITE(*,23) I
      READ(*,*) DALPHA(I)
      WRITE(3,110) I,DALPHA(I)
      ALPHA(I)=DTR*DALPHA(I)
      SA=SIN(ALPHA(I))
      IF(SA.GE.SAMX) SAMX=SA
   33 CONTINUE
      IF(LEVEL.EQ.'2') THEN
      WRITE(*,119)
         READ(*,*) NT
      ELSE
         WRITE(*,20)
         READ(*,*) NT
      ENDIF
      IF(LEVEL.EQ.'2') THEN
         WRITE(*,120)
         READ(*,*) NP
      ELSE
         WRITE(*,21)
         READ(*,*) NP
      ENDIF
      DO 44 I=1,NP
      WRITE(*,24) I
      READ(*,*) DPHI(I)
   44 PHI(I)=DTR*DPHI(I)
      MSTART=0
```

```
      MSTOP=INT(RMAX*SAMX)+1
      WRITE(*,121) MSTOP
      READ(*,*) MSTOP
      IF(SAMX.LT.1.E-6) MSTART=1
C     SETTING RADIAL STEPS BETWEEN RA to R1 AND R2 to RB
      NR1=1
      NR2=1
C     DEFINING OUTSIDE PERMITTIVITY WITH SLIGHT LOSS
      UR(NLAY+1)=(1.0,-1.0E-6)
      ER(NLAY+1)=(1.0,-1.0E-6)
C     ESTIMATING NUMBER OF CORE AND EXTERIOR MODES
      N1=INT(RMIN*CABS(CSQRT(2.0*ER(1)*UR(1))))+1
      N2=INT(RMAX*CABS(CSQRT(2.0*ER(NLAY)*UR(NLAY))))+1
      WRITE(*,122) N1
      READ(*,*) N1
      WRITE(*,123) N2
      READ(*,*) N2
      WRITE(*,*) BELL
      PAUSE 'Check that N1+N2 is .LE. 35 .... Otherwise Abort'
      IF((N1+N2).GT.35) STOP
      WRITE(3,105) MSTART,MSTOP,N1,N2
      WRITE(*,124)
      M1=MSTART+1
      M2=MSTOP+1
      MM=MSTOP-MSTART-1
      IF(MM.LE.0) MM=1
      DO 77 MD=M1,M2
      OPEN(1,FILE='RS.DAT',STATUS='UNKNOWN',ACCESS='SEQUENTIAL',
     1      FORM='UNFORMATTED')
      DD=DMAX-DMIN
      M=MD-1
      WRITE(3,101) FF
      WRITE(3,204) M
      WRITE(*,*) BELL
      WRITE(*,125) M
      BC=0
      IF (M.EQ.1) BC=1
      MF=0
      IF(M.EQ.0) MF=1
C     COMPUTING MODAL EXPANSION TRUNCATION LIMITS
      NM1=N1+1-M-MF
      IF(NM1.LT.1) NM1=1
      NM2=N2+1-M-MF
      IF(NM2.LT.1) NM2=1
      NCOEF=2*(NM1+NM2)
      NMODES=NCOEF+NINC
C     COMPUTING MESH PARAMETERS AND RAD(I) FOR GIVEN MODE = M
      DENS=DMAX-DD*(M+MF-1.0)/MM
      WRITE(*,126)
      CALL MESH (DENS,ER,UR,NR1,NR2,IMX,NDS,LMX,NB,DR,DT,
     1            RA,R1,R2,RB,RMIN,RMAX,NLAY)
C     CHECKING MAXIMUM MESH PARAMETERS
      IF(NB.LE.NBMX) GO TO 55
      WRITE(3,104)
      STOP
   55 CONTINUE
C     LOADING AND MARCHING ((R)) AND ((S))
      WRITE(3,101) FF
      WRITE(3,201)
      WRITE(*,127) IMX
      DO 66 I=1,IMX
      IR=IMX-I
      WRITE(*,128) I,IR
      WRITE(*,129)
      CALL LODER (I,M,BC,ER,UR,NMODES,NM1,NM2,NINC,NB,IMX,IBIG,
     1            NDS,RA,RB,R1,R2,DR,DT,NR1,NR2,RAD,NLAY)
```

```
      IF(I.GT.1) GO TO 57
      WRITE(3,300) I,(RAD(1,L),L=1,NLAY)
 57   CONTINUE
      WRITE(3,300) I,(RAD(2,L),L=1,NLAY)
      WRITE(*,130)
      CALL MARCH (I,IMX,BC,NB,NBMX,NMODES,NMMAX,R,S,RINV)
  66  CONTINUE
      WRITE(*,131)
      CALL SWEEP (M,IMX,BC,NM1,NM2,NINC,NMODES,NMMAX,NCOEF,NCMAX,
     1    NR1,NR2,NB,NBMX,R1,R2,DT,ER,UR,R,S,SINV,FMOM,COEF,NINMX)
      WRITE(*,132)
      CALL MODECO (NM1,NM2,NINC,NCOEF,NCMAX,FMOM,COEF,M,NINMX)
      WRITE(*,133)
      CALL RADFLD (M,NM1,NM2,NT,NP,NA,NCMAX,COEF,NINMX)
      WRITE(*,134)
      CALL CRSSEC (M,NM1,NM2,NA,NCMAX,COEF,NINMX)
      REWIND 10
      READ(10,138) IBIG,IDUMMY
      READ(10,103) RMIN,RMAX
  77  CONTINUE
      CLOSE(1,STATUS='DELETE')
      CLOSE(2)
      CLOSE(10)
      WRITE(*,135)
      CALL DATOUT (MSTART,MSTOP,NT,NP,NA)
      WRITE(*,*) BELL
      WRITE(*,136)
 10   FORMAT(//////////////////7X,'****** WELCOME TO EMCAD ******',//7X,
     1'THIS PROGRAM COMPUTES EM FIELD SCATTERING FROM PENETRABLE',/7X,
     2'BODIES OF REVOLUTION USING THE COUPLED AZIMUTHAL POTENTIAL',/7X,
     3'(CAP) FORMULATION IN CONJUNCTION WITH A VARIATIONAL FINITE-',/7X,
     4'ELEMENT TECHNIQUE AND A TRI-REGIONAL UNIMOMENT METHOD.  THE',/7X,
     5'NUMERICAL SOLUTION IS PERFORMED USING A TWO-SWEEP RICCATI',/7X,
     6'TRANSFORM WITH CIRCUMFERENTIAL MARCHING.',/7X,
     7'UPDATED VERSION OF ORIGINAL U.C. BERKELEY CODE.',/7X,
     8'MODIFICATIONS BY M.A. MORGAN    MAR 1987 - FEB 1988',/7X,
     9'                 E.M. CONNOLLY  JUL 1987 - APR 1988 .',//////7X,
     1'PLEASE PRESS ANY KEY TO CONTINUE.',/)
 12   FORMAT(/////////////////7X,'******* EMCAD INPUT *******',//7X,
     1'EMCAD ALLOWS A USER TO INPUT THE NECESSARY INFORMATION',/7X,
     2'ACCORDING TO THE LEVEL OF HIS EXPERTISE.',//7X,
     3'1 NOVICE LEVEL - GIVES BRIEF EXPLANATIONS/DESCRIPTIONS',/7X,
     4'                 OF INPUT VALUES. DESCRIBES THE REQUIRED',/7X,
     5'                 FORMAT FOR THE INPUT VALUE. GIVES ',/7X,
     6'                 TYPICAL VALUES WHERE APPLICABLE.',//7X,
     7'2 EXPERT LEVEL - ASSUMES THE USER IS FAMILIAR WITH THE',/7X,
     8'                 INPUT PARAMETERS AND FORMATS. SIMPLY',/7X,
     9'                 PROMPTS FOR REQUIRED INPUTS.',//7X,
     1'3 EXIT EMCAD',//7X,
     2'PLEASE SELECT THE LEVEL OF EXPERTISE BY ENTERING 1, 2, OR 3.',
     3//7X)
  13  FORMAT(////////////////////////7X,
     1'THE OUTPUT DATA FILE IS THE FILE CONTAINING THE OUTPUT',/7X,
     2'RESULTS OF ALL NUMERICAL CALCULATIONS CONDUCTED BY EMCAD.',/7X,
     3'THE FORMAT FOR THIS INPUT IS FILENAME ONLY. NO EXTENSION IS',/7X,
     4'REQUIRED OR DESIRED. EMCAD AUTOMATICALLY APPENDS AN ',/7X,
     5'EXTENSION OF .OUT TO YOUR FILENAME.',//7X,
     6'PLEASE ENTER THE FILENAME OF THE OUTPUT DATA FILE.',//7X)
  14  FORMAT(////////////////////////7X,
     1'THE OUTPUT GRAPHICS DATA FILE IS THE FILE CONTAINING THE ',/7X,
     2'OUTPUT DATA FROM EMCAD TO BE USED AS INPUT TO GRAPHING ',/7X,
     3'ROUTINES. THE FORMAT FOR THIS INPUT IS FILENAME ONLY.  NO',/7X,
     4'EXTENSION IS REQUIRED OR DESIRED.   EMCAD AUTOMATICALLY ',/7X,
     5'APPENDS AN EXTENSION OF .TMT, .TMP, .TET, AND .TEP TO YOUR',/7X,
     6'FILENAME AS IT PRODUCES FOUR OUTPUT FILES FOR GRAPHICS.',//7X,
     7'.TMT -----> TM INCIDENCE, F-THETA',/7X,
```

```
     8'.TMP -----> TM INCIDENCE, F-PHI',/7X,
     9'.TET -----> TE INCIDENCE, F-THETA',/7X,
     1'.TET -----> TE INCIDENCE, F-PHI',//7X,
     2'PLEASE ENTER THE FILE NAME OF THE OUTPUT DATA FILE.',//7X)
  15 FORMAT(/////////////////////////7X,
     1'THE GRAPHICS CAPTION IS A PERSONALIZED CAPTION ALLOWING THE',/7X,
     2'USER TO IDENTIFY THIS SET OF GRAPHS FROM ALL OTHER SETS.',/7X,
     3'THE MAXIMUM LENGTH OF THIS CAPTION IS 64 CHARACTERS.',/7X,
     4'NOTE: WHEN USED WITH THE GRAPHICS PACKAGE, THE PROGRAM IS',/7X,
     5'ABLE TO DIFFERENTIATE BETWEEN UPPER CASE AND LOWER CASE',/7X,
     6'CHARACTERS.',//7X,
     7'PLEASE ENTER ANY GRAPHICS CAPTION YOU DESIRE',//7X)
  16 FORMAT(////////////////////////7X,
     1'DMIN AND DMAX ARE PARAMETERS OF MESH DENSITY IN TERMS OF',/7X,
     2'ELEMENTS/INTERIOR LAMBDA. INPUT VALUES ARE EXPECTED TO BE',/7X,
     3'REAL, I.E. THE DECIMAL POINT MUST BE INCLUDED.',/7X,
     4'TYPICAL VALUES ARE DMIN = 10. AND DMAX = 15.',//7X,
     4'PLEASE INPUT DMIN',//7X)
  17 FORMAT(//7X,'PLEASE INPUT DMAX',//7X)
  18 FORMAT(////////////////////////7X,
     1'THE NUMBER OF INCIDENT FIELD ANGLES IS THE TOTAL NUMBER OF',/7X,
     2'INCIDENT FIELDS THAT IMPINGE ON THE OBJECT OF INTEREST.',/7X,
     3'THIS PROGRAM ALLOWS A MAXIMUM OF FIVE INCIDENT FIELD ',/7X,
     4'ANGLES.  WHEN ENTERING YOUR ANSWER PLEASE DO NOT INCLUDE A',/7X,
     5'DECIMAL BECAUSE THE INPUT MUST BE IN INTEGER FORMAT, I.E.',//7X,
     5'           NA = 3   ,  NA = 1  ',//7X,
     6'PLEASE INPUT THE NUMBER OF INCIDENT FIELD ANGLES.',//7X)
  19 FORMAT(////////////////////////7X,
     1'THE INCIDENT ANGLES DEFINE THE DIRECTION FROM WHICH THE ',/7X,
     2'INCIDENT FIELDS ORIGINATE.',//7X,
     3'0 DEGREES  -----> + Z DIRECTION',/7X,
     4'90 DEGREES  -----> + X DIRECTION',/7X,
     5'180 DEGREES  -----> - Z DIRECTION',//7X)
  20 FORMAT(////////////////////////7X,
     1'THE NUMBER OF SCATTERING FIELD THETA POINTS DETERMINES THE',/7X,
     2'SPACING BETWEEN THETA POINTS DURING EMCAD ITERATIONS AND',/7X,
     3'CALCULATIONS.',//7X,
     4'DELTA THETA = 180 / (NUMBER THETA POINTS - 1)  SO...',//7X,
     5'NUMBER THETA POINTS = 37  -----> DELTA THETA = 5 DEGREES',//7X,
     6'NUMBER THETA POINTS = 19  -----> DELTA THETA = 10 DEGREES',//7X,
     7'WHEN ENTERING YOUR ANSWER, PLEASE DO NOT INCLUDE A DECIMAL',/7X,
     8'BECAUSE THE INPUT MUST BE IN INTEGER FORMAT. I.E. NT = 19',///7X,
     9'PLEASE INPUT THE NUMBER OF SCATTERING FIELD THETA POINTS',//7X)
  21 FORMAT(////////////////////////7X,
     1'THE NUMBER OF PHI ANGLES IS THE TOTAL NUMBER OF PHI ANGLES.',/7X,
     2'THIS PROGRAM ALLOWS A MAXIMUM OF EIGHT PHI ANGLES.  WHEN ',/7X,
     3'ENTERING YOUR ANSWER PLEASE DO NOT INCLUDE A DECIMAL ',/7X,
     4'BECAUSE THE INPUT MUST BE IN INTEGER FORMAT.  I.E.',//7X,
     5'     NP = 3   ,  NP = 1  ',//7X,
     6'PLEASE INPUT THE NUMBER OF PHI ANGLES.',//7X)
  23 FORMAT(/7X,'ENTER INC FLD ANGLE (DEG) FOR # ',I3,//7X)
  24 FORMAT(/7X,'ENTER PHI FLD ANGLE (DEG) FOR # ',I3,//7X)
  25 FORMAT(//////7X,'PLEASE PRESS ANY KEY TO CONTINUE.',////)
 100 FORMAT(/,'******************** EMCAD OUTPUT DATA ****************
     1************')
 101 FORMAT(A)
 102 FORMAT(I5)
 103 FORMAT(4(E14.6))
 104 FORMAT(' *** PROGRAM ABORTED BECAUSE NB > NBMX ***')
 105 FORMAT(/7X,'SERIES SOLUTION PARAMETERS'/7X,'MSTART=',I5,/7X,
     1'MSTOP= ',I5,/7X,'N1= ',I8,/7X,'N2= ',I8)
 107 FORMAT(//7X,I5,2(2X,'(',1PE11.3,2X,1PE11.3,')'))
 109 FORMAT(//7X,'INCIDENT FIELD ANGLES',//9X,'N',10X,'ALPHA(N)')
 110 FORMAT(//7X,I3,7X,F5.0,'  DEG')
 111 FORMAT(/7X,'COMPLEX Er(n) AND Ur(n)')
 112 FORMAT(//7X,'ENTER STRUCTURE DATA FILE (D:FILENAME.EXTENSION):  ')
```

```
  113 FORMAT(//7X,'ENTER OUTPUT DATA FILENAME WITHOUT EXTENSION',//7X,
     1'FOR PRINTER OUTPUT ENTER "LPT1":      ')
  114 FORMAT(//7X,'ENTER GRAPHICS DATA FILENAME WITHOUT EXTENSION    ')
  115 FORMAT(//7X,'ENTER CAPTION OR LABEL      ')
  116 FORMAT(//7X,'ENTER DMIN (ELEMENTS/INTERIOR LAMBDA):    ')
  117 FORMAT(//7X,'ENTER DMAX (ELEMENTS/INTERIOR LAMBDA):    ')
  118 FORMAT(//7X,'ENTER NUMBER OF INCIDENT FIELD ANGLES (.LE. 5)   ')
  119 FORMAT(//7X,'ENTER NUMBER OF THETA POINTS FOR SCAT FIELD (.LE 37.)
     1',4X)
  120 FORMAT(//7X,'ENTER NUMBER OF PHI ANGLES (.LE. 8):   ')
  121 FORMAT(//7X,'ENTER MSTOP (.LE. 13)',//7X,
     1'ESTIMATED "MINIMUM" VALUE IS:',4X,I4,//7X)
  122 FORMAT(//7X,'ENTERING NO. OF INTERNAL AND EXTERNAL EXPANSION MODES
     1',//7X,
     2'ESTIMATED (KI*RMIN) "MINIMUM" INTERNAL N1 =',2X,I4,//7X,
     3'ENTER N1  (.GE. MSTOP):  ',//7X)
  123 FORMAT(//7X,'ESTIMATED (KO*RMIN) "MINIMUM" EXTERNAL N2 =',2X,I4,
     1//7X,'ENTER N2  (.GE. MSTOP):  ',//7X)
  124 FORMAT(//7X,'INDEXING PROGRAM THROUGH VALUES OF M')
  125 FORMAT(7X,'M-LOOP .... M =  ',I5)
  126 FORMAT(7X,'CALL MESH')
  127 FORMAT(7X,'ENTER I-LOOP;   NO STEPS TO COMPLETE:  ',I6)
  128 FORMAT(7X,'I=',2X,I5,7X,'NO STEPS TO GO:',2X,I5)
  129 FORMAT(7X,'CALL LODER')
  130 FORMAT(7X,'CALL MARCH')
  131 FORMAT(7X,'EX I-LOOP, SWEEP')
  132 FORMAT(7X,'CALL MODECO')
  133 FORMAT(7X,'CALL RADFLD')
  134 FORMAT(7X,'CALL CRSSEC')
  135 FORMAT(7X,'EX M-LOOP, DATOUT')
  136 FORMAT(//7X,'*************** EMCAD COMPLETED **************')
  137 FORMAT(//7X,'ENTER MATERIAL PARAMETER FILE (D:FILENAME.EXTENSION)
     1:  ')
  138 FORMAT(2(I5))
  201 FORMAT(//7X,' RICCATI MARCHING STEPS')
  204 FORMAT(//7X,' ************** M = ',I5,' **************'//)
  300 FORMAT(7X,I5,5E12.3)
   99 STOP
      END
```

The above constitutes only the main program of EMCAD. The remaining subroutines would take approximately 26 more pages to print. The entire program code can be obtained by sending an IBM PC compatible formatted 5 1/4 " floppy disk to

Prof. M. A. Morgan
Code 62 MW
Naval Postgraduate School
Monterey, Ca. 93943

APPENDIX D

EMCADIN USER'S MANUAL

Welcome to EMCADIN. This program is designed to accept properly formatted input data from the West Coast Consultants commercial shareware product CURVE DIGITIZER (Appendix M) and convert it to a form which can be used by the scattering calculation program EMCAD. CURVE DIGITIZER is a computer aided design (CAD) package which will allow user manipulation of the data.

A. BACKGROUND

EMCADIN was written in the programming language of Ryan-McFarland FORTRAN (Appendix M). Any questions concerning formats, changes to the program, etc... should be referred to the respective user's manuals, Dr. M. A. Morgan or LT. E. M. Connolly (Appendix M).

B. COMPUTER SETUP

EMCADIN was not written for any specific computer configuration, and as such, should run on most IBM compatible machines.

C. PROGRAM INPUT AND OUTPUT

EMCADIN obtains its input from a previously generated data file. The input data file for EMCADIN must be located in the same directory/subdirectory as EMCADIN. This data

file is produced through the utilization of the computer aided design package, CURVE DIGITIZER. As shown in Figure D-1, the output of CURVE DIGITIZER is the input to EMCADIN. For EMCADIN and CURVE DIGITIZER to be compatible, it was necessary to establish and adhere to the predefined data format of CURVE DIGITIZER. In order for EMCADIN to be utilized successfully, it is imperative that the user understand the need for the particular format being used, as well as the format itself. The transition from CURVE DIGITIZER to EMCADIN provides more of an opportunity for catastrophic error than at any other point in this entire package of interactive computer programs.

Understanding the format of the input of the data for EMCADIN necessitates a thorough understanding of some of the idiosyncracies of CURVE DIGITIZER, referred to henceforth for brevity as CD. Computer aided graphic design deals in the basic business of handling points. Any graphic figure can be represented as some combination of points. Lines can be formed by connecting two points. Arcs can be represented by placing more points closer together and by approximating the curvature with sufficiently small line segments between points. The required input-output format for CD is utilized to insure that adequate information is present to direct CD how to connect the points. A similar process may be that of the simple children's game connect-the-dots. A series of dots exists which define a picture,

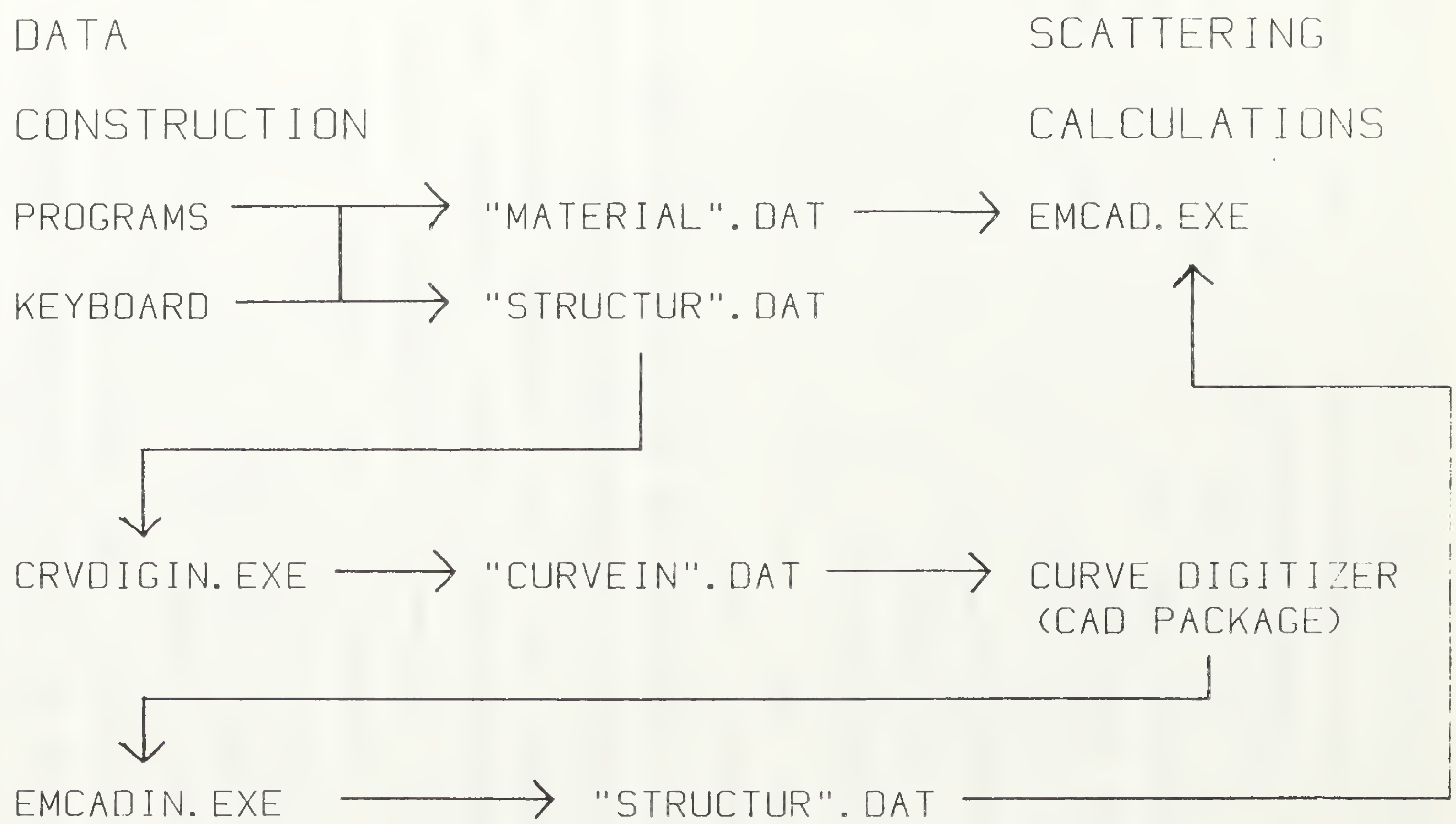


FIG. D-1. CRVDIGIN AND EMCADIN FLOW DIAGRAM

if connected properly. As more and more dots are used, more complex pictures with better resolution can be constructed. The fundamental assumption employed by CD is that each point should be connected to the next point listed in the data file unless some code is present to indicate otherwise. For simplicity and unrestricted accessibility, CD input and output data are represented in terms of XY rectangular coordinates. CD flags the break between two data points by what shall be referred to as a separation code. Since two coordinates, X and Y, are required to establish a data point, two separation codes, one for the X data and one for the Y data are required to separate two distinct data points. The separation code employed is of the form 999990. The last two digits, 90, may change to reflect scaling, color, or any of a number of other things. It was not possible to determine the specifics from the user manual. In the specific case of EMCADIN input data, 999990 999990 codes are required to distinguish the structural data describing one layer from that describing another layer.

CD stores the data in the exact order in which the data points were drawn. There are other methods of generating the simple geometric shapes demonstrated in the next two examples, however, these examples were specifically chosen to illustrate potential pitfalls that could occur if extreme care is not taken. For the first example, the simple square

of Figure D-2 was constructed utilizing CD. The base of the square was drawn by defining point number 1, and then by drawing a line to point number 2. The top was likewise established by defining point number 4, and then by drawing a line to point number 3. The two sides were then formed by connecting point number 4 to point number 1 with a line, and by connecting point number 2 to point number 3 with a line. This is one of many methods of connecting the four data points that define this square. Figure D-3 lists the output data file which contains the CD defining data for this square. Notice there are significantly more than four data points describing this object which requires only four data points. Recall from earlier discussion that EMCADIN utilizes the 999990 codes to distinguish structural data describing one layer from that describing another layer. Figure D-2 depicts only one layer, yet EMCADIN using the data file of Figure D-3 would interpret the data as four distinct layers. The 999990 codes of Figure D-3 are present due to the order in which the data points were entered. Had the square been generated by defining point number 3, connecting it to point number 2, then to 1, then to 4, there would have been no 999990 codes separating the data, and the data file of Figure D-3 would consist of only four pairs (XY coordinates) of data.

The second and third major errors that might occur can also be demonstrated using Figure D-2. The sole

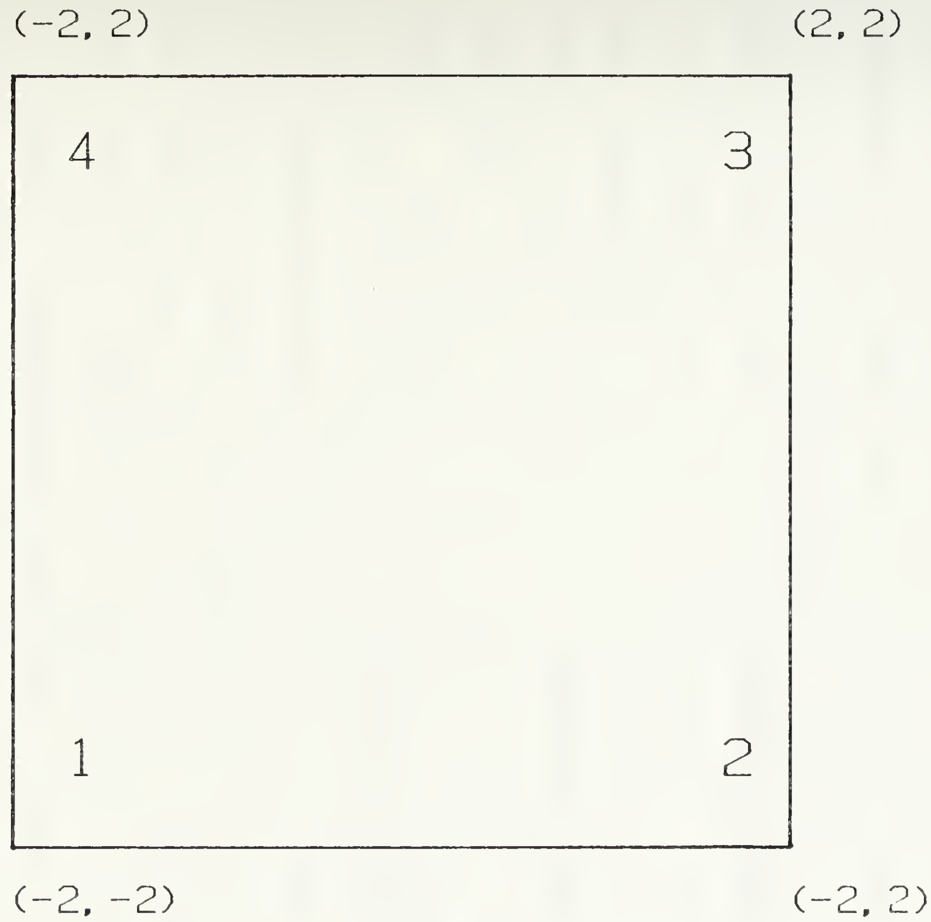


FIG. D-2. EXAMPLE OF CD GENERATION OF A SIMPLE SQUARE

```
-2,-2
 2,-2
 999990, 999990
-2, 2
 2, 2
 999990, 999990
-2, 2
-2,-2
 999990, 999990
 2,-2
 2, 2
END,END
-5.5           5.5           -4.5           4
 11            8.5
```

FIG. D-3. EXAMPLE OF AN OUTPUT DATA FILE FOR A CD GENERATED SQUARE

justification for utilizing CD was its ability to generate data to be used ultimately by the scattering program EMCAD. The main body of the thesis discusses the requirement of EMCAD for axisymmetric bodies of revolution. Any three dimensional axisymmetric body of revolution can be defined by knowing the axis of revolution, and then by defining the body in planar terms. As an illustration, a given sphere is a three dimensional body of revolution. The sphere has axial symmetry and can be described by a semi-circle in any meridian plane intersecting the axis. Rotating this planar semicircle about the axis would produce the original sphere. EMCAD requires this meridian representation of the three dimensional body as its input. For simplicity's sake, the CD representation of the meridian plane will be referred to as the XY plane, with X being the horizontal axis and Y being the vertical axis, as well as the axis of revolution. This coordinate system is for discussion purposes only, and is not to be confused with the three dimensional coordinate system of EMCAD theory. Viewing the body of Figure D-2 as a two dimensional body, the Y axis is the required axis of revolution. Therefore all CD planar representations for EMCAD should be constructed with values on the X axis restricted to positive values.

The third problem occurs when removal of data points is attempted in order to obtain a proper planar representation. EMCADIN performs a valuable service to the user of improving

the resolution of input data. It suffices to say that due to the inherent mathematics involved, EMCAD requires that two data points be defined, both on the Y axis. The first data point defines the beginning of the object, at X = 0, and the last data point defines the end of the object, also at X = 0. In the example of the square, the beginning data point would be (0,2), and the ending data point would be (0,-2). Without these beginning and ending data points for each layer of data, the EMCADIN interpolation will "bomb", and will produce erroneous results, if any results at all.

For the second example, the simple circle of Figure D-4 was chosen. CD allows the user to construct a circle by simply defining the coordinates of the center of the circle, and by specifying the desired radius. In every instance, CD generates the data points by starting at point 3 on Figure D-4, progressing to point 2, to point 5, to point 4, and back to point 3 in a counter clockwise pattern. A sample of the output data file for this circle is shown in Figure D-5. This data, for a chosen radius of 2, reflects the same counter clockwise pattern, which is directly opposite to the pattern required by EMCADIN input data. In EMCADIN, theta is defined in a clockwise pattern, starting on the positive y axis and progressing by equal step sizes clockwise to the negative y axis. By using this method of circle construction, the user must find someway to edit and rearrange the data before it can be used by CD. The other

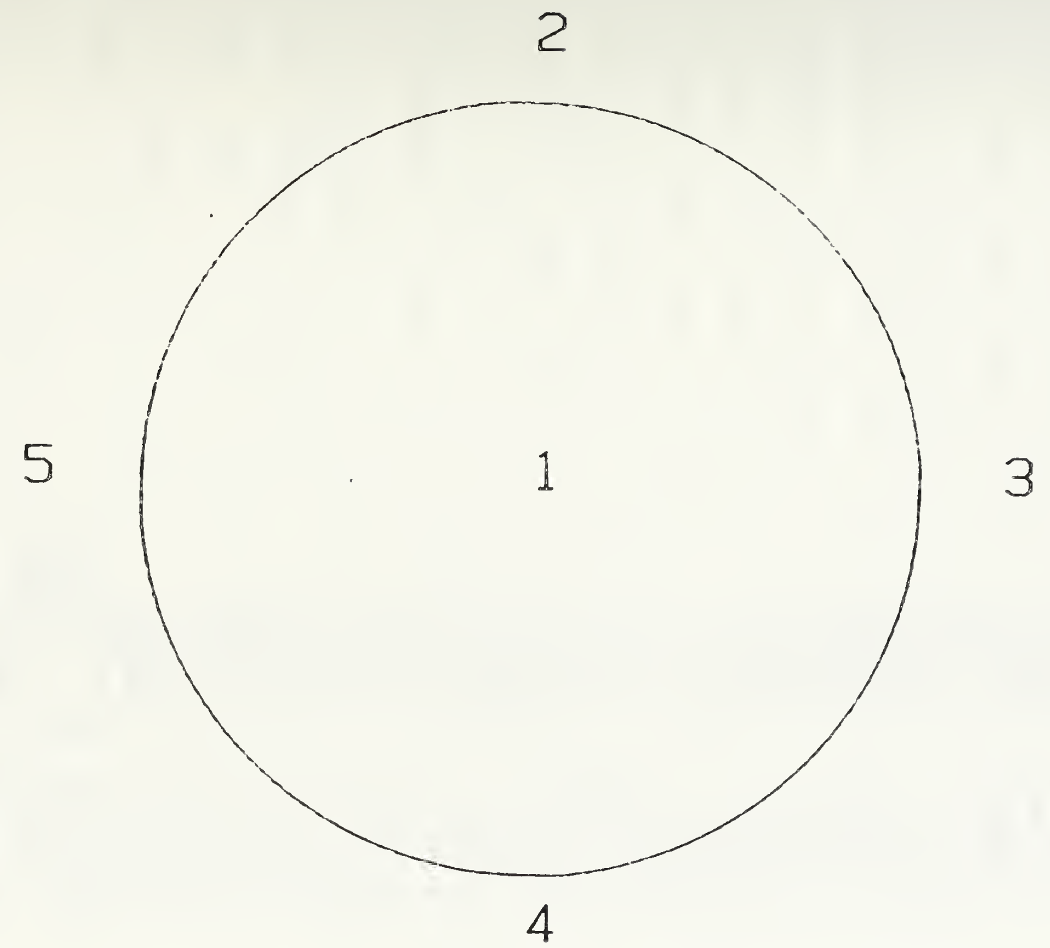


FIG. D-4. EXAMPLE OF CD GENERATION OF A SIMPLE CIRCLE

```
 2, Ø
 1.997964, .Ø9Ø2255
 1.99186, .18Ø2673
 1.981699, .269942
 1.9675Ø4, .359Ø671
 1.9493Ø2, .447461
 1.927131, .5349439
 1.9Ø1Ø37, .6213375
 1.871Ø71, .7Ø64659
 1.837295, .79Ø1558
 1.799778, .8722369
 1.758597, .9525418
 1.713835, 1.Ø3Ø9Ø7
 1.665583, 1.1Ø7173
 1.61394, 1.181185
 1.559Ø1, 1.252792
 1.5ØØ9Ø6, 1.321848
 1.439746, 1.388212
 1.375654, 1.451749
 1.3Ø8761, 1.512331
 1.2392Ø3, 1.569833
 1.167122, 1.624138
 1.Ø92665, 1.675137
 1.Ø15982, 1.722725
 .93723Ø9, 1.7668Ø5
 .8565713, 1.8Ø7287
 .7741675, 1.844Ø89
 .69Ø1873, 1.877137
 .6Ø48Ø18, 1.9Ø6362
 .5181847, 1.9317Ø5
 .43Ø5126, 1.953115
 .3419638, 1.97Ø549
 .2527188, 1.983969
 .1629591, 1.99335
 7.286744E-Ø2, 1.998672
-1.737241E-Ø2, 1.999925
-.1Ø75769, 1.9971Ø5
-.1975623, 1.99Ø218
```

FIG. D-5. EXAMPLE OF A PORTION OF AN OUTPUT DATA FILE FOR A CD GENERATED CIRCLE

problem that arises, particularly with circles, is that as the radius becomes smaller and smaller, fewer and fewer points are used to construct the circle, causing the data points to become less and less accurate.

For the case of multiple layered bodies, EMCADIN expects that the data point values for each layer have been entered in the input data file starting with the layer closest to the center of the object, labeled as Layer 1 in Figure D-6, then advancing outward from the center until all layers have been accounted for. It has already been stated that data points are listed in the order in which they are entered, which means that not only might data points in a given layer be out of proper order, but data points in differing layers, or even the layers themselves might be interspersed. At the present time, there appears to be only two methods to preclude the abundance of possible errors. The first of these dictates the use of extreme care when entering the data points. The values must be entered in order, starting at $\theta = 0$ degrees and progressing clockwise to $\theta = 180$ degrees for the layer closest to the center point, and then entered for the next closest layer, and then the next closest layer ... until all layers have been described. Even a small band defined by a different material parameter must be defined for all θ from 0 degrees to 180 degrees. If this process is followed exactly, there is no need for the second method. The second method

FIG. D-6. GRAPHICAL REPRESENTATION OF EMCADIN VIEW OF LAYERED DATA

involves the use of some text editor to rearrange the data as necessary, remove unnecessary 999990 codes, insure data for layers as well as layers are in the proper order, etc... The prudent method would include extreme caution coupled with a data review utilizing a text editor.

The EMCADIN program output data consists of the number of layers of the body being considered, the minimum and maximum values of radius, and the listing of radius values in a clockwise manner for $\theta = 0$ to $\theta = 180$ degrees for each layer.

D. PROGRAM EXECUTION

Once the correct disk or directory has been selected, EMCADIN can be invoked by typing

EMCADIN

at the DOS prompt. The program will load itself and the user will see the following screens:

* Welcome to EMCADIN (Figure D-7)

This screen presents some brief background information.

* Input Filename (Figure D-8)

This screen prompts the user for the name of the file containing the input data for EMCADIN. If the file is obtained from CD, the file must have an extension of .dat,

WELCOME TO EMCADIN

THIS PROGRAM ACCEPTS PROPERLY FORMATTED INPUT DATA
FROM CURVE DIGITIZER AND CONVERTS IT TO A FORM
WHICH CAN BE USED BY EMCAD.

PLEASE PRESS ENTER TO CONTINUE.

FIG. D-7. WELCOME TO EMCADIN MENU

INPUT FILENAME

INPUTFN IS THE INPUT FILENAME OF THE FILE CONTAINING THE DATA OF INTEREST WHICH WILL BE INTERPOLATED ON AND CONVERTED TO A FORM WHICH CAN SUBSEQUENTLY BE USED BY EMCAD.

PLEASE ENTER THE NAME OF THE INPUT FILE. THE EXTENSION OF THE FILENAME MUST BE .DAT, I.E. FILENAME.DAT

FIG. D-8. EMCADIN PROMPT FOR INPUT FILENAME

and the extension must be entered by the user as part of the filename. The input data file must be in the same directory/subdirectory as EMCADIN.

* Output Filename (Figure D-9)

This screen prompts the user for the name of the file that will hold the output data from EMCADIN. The extension of this file is left to user discretion, but must be included as part of the output filename.

* Resolution (Figure D-10)

EMCADIN uses linear interpolation to provide the capability to generate output data with better resolution than the input data. This screen prompts the user for the desired output data resolution, in the units of degrees.

OUTPUT FILENAME

OUTPUTFN IS THE OUTPUT FILENAME OF THE FILE CONTAINING THE DATA WHICH HAS BEEN INTERPOLATED AND CONVERTED TO A FORM WHICH WILL BE USED BY EMCAD.

PLEASE ENTER THE NAME OF THE OUTPUT FILE. THE EXTENSION OF THE FILENAME MUST BE INCLUDED I.E. FILENAME.DAT

FIG. D-9. EMCADIN PROMPT FOR OUTPUT FILENAME

RESOLUTION

DELTHE IS THE USER INPUT VALUE OF THE DESIRED THETA
RESOLUTION IN DEGREES.

PLEASE ENTER THE DESIRED DELTA THETA VALUE IN DEGREES.

FIG. D-10. EMCADIN PROMPT FOR OUTPUT DATA RESOLUTION

APPENDIX E

EMCADIN SOURCE CODE

```
C       PROGRAM EMCADIN
C
C       ******************************************************************
C       WELCOME TO EMCADIN WRITTEN BY LT  E. M. CONNOLLY.
C       EMCADIN IS A PROGRAM DESIGNED TO ACCEPT PROPERLY FORMATTED
C       INPUT DATA FROM CURVE DIGITIZER AND TO CONVERT IT TO A
C       FORM WHICH CAN BE USED BY THE PROGRAM EMCAD, WRITTEN BY
C       DR.  M. A. MORGAN.
C
C       ******************************************************************
C       VARIABLE DEFINITIONS
C       ******************************************************************
C
C       INPUTFN           THE INPUT FILENAME OF THE FILE CONTAINING
C                         THE DATA OF INTEREST WHICH WILL BE INTERPOLATED
C                         ON AND CONVERTED TO A FORM WHICH CAN
C                         SUBSEQUENTLY BE USED BY EMCAD
C       OUTPUTFN          THE OUTPUT FILENAME OF THE FILE CONTAINING
C                         THE DATA WHICH HAS BEEN INTERPOLATED AND
C                         CONVERTED TO A FORM WHICH WILL BE USED BY
C                         EMCAD
C       I, J, K,          INTEGER COUNTERS
C       L, N
C       RADMIN            THE MINIMUM RADIUS VALUE
C       RADMAX            THE MAXIMUM RADIUS VALUE
C       PI                A CONSTANT
C       DELTHE            THE USER INPUT VALUE OF THE DESIRED THETA
C                         RESOLUTION IN DEGREES
C       DELTHETA          THE USER INPUT VALUE OF THE DESIRED THETA
C                         RESOLUTION IN RADIANS
C       THETA(N)          AN ARRAY HOLDING THE DESIRED VALUES OF THETA
C                         AS DETERMINED BY DELTHE
C       DEGTORAD          A CONVERSION FACTOR FROM DEGREES TO RADIANS
C       NUMPTSIN          THE NUMBER OF POINTS HELD IN A GIVEN INPUT
C                         DATA SET
C       NUMSETS           THE NUMBER OF INPUT DATA SETS
C       DUMMY1            AN INPUT VARIABLE ALLOWING SCREEN ADVANCING
C                         DURING MENU SELECTION
C       DUMMY2            A CHARACTER STRING USED TO LOCATE THE END OF
C                         EACH INPUT DATA SET
C       XARRAY(I)         AN ARRAY CONTAINING THE VALUES OF X FOR ALL
C                         THE INPUT DATA SETS
C       YARRAY(I)         AN ARRAY CONTAINING THE VALUES OF Y FOR ALL
C                         THE INPUT DATA SETS
C       A(I,J)            AN ARRAY CONTAINING THE VALUES OF X FOR ALL
C                         THE J INPUT DATA SETS
C       B(I,J)            AN ARRAY CONTAINING THE VALUES OF Y FOR ALL
C                         THE J INPUT DATA SETS
C       NUMPTOUT          THE NUMBER OF POINTS OUT, DETERMINED BY THE
C                         DESIRED RESOLUTION
C       THETAI            THE VALUE OF THETA DETERMINED BY THE I-TH INPUT
C                         X AND Y VALUES
C       THETAK            THE VALUE OF THETA DETERMINED BY THE K-TH INPUT
C                         X AND Y VALUES
C       XFOUND            THE DESIRED VALUE OF X CORRESPONDING TO A GIVEN
C                         VALUE OF THETA(N)
C       YFOUND            THE DESIRED VALUE OF Y CORRESPONDING TO A GIVEN
C                         VALUE OF THETA(N)
```

```
C     RADIUS(N,J)        AN ARRAY CONTAINING THE OUTPUT CALCULATED
C                        VALUES OF RADIUS FOR ALL THE INPUT DATA SETS
C     M                  SLOPE OF THE LINE CONTAINING THE I-TH AND K-TH
C                        DATA POINTS
C     BB                 Y-INTERCEPT OF THE LINE CONTAINING THE I-TH AND
C                        K-TH DATA POINTS
C
C     *****************************************************************
C     VARIABLE DECLARATIONS
C     *****************************************************************
      DOUBLE PRECISION XINPUT(2000), YINPUT(2000), RADIUS(2000,5)
      DOUBLE PRECISION DELTHE, DELTHETA, THETA(2000), THETAI, THETAK
      DOUBLE PRECISION A(2000,5), B(2000,5)
      DOUBLE PRECISION DEGTORAD, INTER1, INTER2, RADMIN, RADMAX
      DOUBLE PRECISION PI, DUMMY, XFOUND, YFOUND
      DOUBLE PRECISION BB, M
      INTEGER I, J, K, L, N, NUMPTSIN, NUMPTOUT, NUMSETS
      CHARACTER*3 DUMMY1, DUMMY2
      CHARACTER*12 INPUTFN, OUTPUTFN
C
C     *****************************************************************
C     INITIAL VALUES
C     *****************************************************************
      PI = 3.14159
      J = 0
      K = 0
      L = 0
      NUMPTSIN = 0
      NUMSETS = 1
      DEGTORAD = PI / 180.
      RADMIN = 10000
      RADMAX = -10000
C
C     *****************************************************************
C     INPUT DATA OF INTEREST
C     *****************************************************************
      WRITE(*,110)
      READ(*,100) DUMMY1
      WRITE(*,120)
      READ(*,100) INPUTFN
      WRITE(*,130)
      READ(*,100) OUTPUTFN
      WRITE(*,140)
      READ(*,*) DELTHE
C
C     ******************************************************************
C
C     ******************************************************************
C     DETERMINE LOCATION OF END OF DATA
C     ******************************************************************
      OPEN(7, FILE = INPUTFN)
      OPEN(8,FILE = OUTPUTFN)
   10 READ (7,100) DUMMY2
      NUMPTSIN = NUMPTSIN + 1
      IF(DUMMY2 .NE. 'END') GOTO 10
      CLOSE(7)
      OPEN(7, FILE = INPUTFN)
      N = 1
      J = 1
      DO 20 I= 1,NUMPTSIN - 1
      READ (7,*) XINPUT(I), YINPUT(I)
      IF(ABS(XINPUT(I) - 999990) .GT. 20.) THEN
      A(N,J) = XINPUT(I)
      B(N,J) = YINPUT(I)
      NUMSETS = J
      ELSE
```

```
      J = J + 1
      N = 0
      ENDIF
      N = N + 1
   20 CONTINUE
C
C     ****************************************************************
C     INTERMEDIATE CALCULATIONS
C     ****************************************************************
      NUMPTOUT = INT(180. / DELTHE) + 1
      DELTHETA = DELTHE * DEGTORAD
C
C     ****************************************************************
C     INTERPOLATION FOR THETA RESOLUTION
C     ****************************************************************
      DO 30 J = 1, NUMSETS
      DO 40 N = 1, NUMPTOUT
C     DETERMINE THETA WHERE DATA POINTS X AND Y ARE TO BE FOUND
      THETA(N) = ( N - 1 ) * DELTHETA
      DO 50 I = 1, NUMPTSIN - 1
      K = I-1
      IF( I .EQ. 1) B(K,J) = B(I,J)
      IF(ABS(B(I,J)) .LT. .001) THEN
      THETAI = PI/2
      ELSE IF(B(I,J) .LT. 0) THEN
      THETAI = PI - DATAN(DABS(A(I,J)/B(I,J)))
      ELSE
      THETAI = DATAN(ABS(A(I,J)/B(I,J)))
      ENDIF
C     DETERMINE IF K-TH DATA PAIR FORMS ANGLE GREATER THAN 90 DEGREES
C     ASSUMES CENTER OF COORDINATE SYSTEM AT X = 0  Y = 0
      IF(ABS(B(K,J)) .LT. .001) THEN
      THETAK = PI/2
      ELSE IF(B(K,J) .LT. 0) THEN
      THETAK = PI - DATAN(DABS(A(K,J)/B(K,J)))
      ELSE
      THETAK = DATAN(ABS(A(K,J)/B(K,J)))
      ENDIF
C     CHECK TO SEE IF K-TH POINT IS THE DESIRED DATA POINT
      IF(DABS(THETA(N) - THETAK) .LT. .001) THEN
      XFOUND = A(K,J)
      YFOUND = B(K,J)
      RADIUS(N,J) = 2*PI*DSQRT(XFOUND**2 + YFOUND**2)
      IF(RADIUS(N,J) .LT. RADMIN) RADMIN = RADIUS(N,J)
      IF(RADIUS(N,J) .GT. RADMAX) RADMAX = RADIUS(N,J)
      GOTO 40
C     CHECK TO SEE IF I-TH POINT IS THE DESIRED DATA POINT
      ELSE IF(DABS(THETA(N) - THETAI) .LT. .001) THEN
      XFO"ND = A(I,J)
      YFOUND = B(I,J)
      RADIUS(N,J) = 2*PI*DSQRT(XFOUND**2 + YFOUND**2)
      IF(RADIUS(N,J) .LT. RADMIN) RADMIN = RADIUS(N,J)
      IF(RADIUS(N,J) .GT. RADMAX) RADMAX = RADIUS(N,J)
      GOTO 40
C     CHECK TO SEE IF THE DESIRED DATA POINT IS BETWEEN THE K-TH
C     DATA POINT AND THE I-TH DATA POINT
      ELSE IF(THETA(N) .GE. THETAK .AND.
     1         THETA(N) .LE. THETAI) THEN
C     CHECK FOR VERTICAL LINE
      IF(DABS(A(I,J)-A(K,J)) .LT. .0001) THEN
      XFOUND = A(I,J)
      YFOUND = XFOUND / DTAN( THETA(N) )
      RADIUS(N,J) = 2*PI*DSQRT(XFOUND**2 + YFOUND**2)
      IF(RADIUS(N,J) .LT. RADMIN) RADMIN = RADIUS(N,J)
      IF(RADIUS(N,J) .GT. RADMAX) RADMAX = RADIUS(N,J)
      GOTO 40
```

```
C     CHECK FOR HORIZONTAL LINE
      ELSE IF(DABS(B(I,J)-B(K,J)) .LT. .0001) THEN
      YFOUND = B(I,J)
      XFOUND = YFOUND * DTAN( THETA(N) )
      RADIUS(N,J) = 2*PI*DSQRT(XFOUND**2 + YFOUND**2)
      IF(RADIUS(N,J) .LT. RADMIN) RADMIN = RADIUS(N,J)
      IF(RADIUS(N,J) .GT. RADMAX) RADMAX = RADIUS(N,J)
      GOTO 40
C     SLOPE OF LINE CONTAINING I-TH AND K-TH DATA POINTS
      ELSE
      M = ( B(I,J)-B(K,J) ) / ( A(I,J)-A(K,J) )
C     Y-INTERCEPT OF LINE CONTAINING I-TH AND K-TH DATA POINTS
      BB = B(I,J) - ( M * A(I,J) )
C     RECALL EQUATION OF ANY LINE IS Y = MX + B
C     THE LINE CONTAINING THE I-TH AND K-TH DATA POINTS INTERSECTS
C     THE LINE DRAWN FROM THE ORIGIN THROUGH THE DESIRED DATA POINT
C     AT THE DESIRED DATA POINT XFOUND AND YFOUND.  THE EQUATION FOR
C     THE LINE CONTAINING THE I-TH AND K-TH DATA POINTS MUST THEN
C     ALSO WORK AT THE DESIRED DATA POINT.  WRITING XFOUND AND
C     YFOUND IN TERMS OF THEIR ANGLE THETA(N), AND SUBSTITUTING
      YFOUND = BB / ( 1 - ( M * DTAN( THETA(N) ) ) )
      XFOUND = YFOUND * DTAN( THETA(N) )
      RADIUS(N,J) = 2*PI*DSQRT(XFOUND**2 + YFOUND**2)
      IF(RADIUS(N,J) .LT. RADMIN) RADMIN = RADIUS(N,J)
      IF(RADIUS(N,J) .GT. RADMAX) RADMAX = RADIUS(N,J)
      GOTO 40
      ENDIF
C     NEED TO TRY A NEW I-TH AND K-TH DATA PAIR
      ELSE
      DUMMY = 0
      ENDIF
   50 CONTINUE
   40 CONTINUE
   30 CONTINUE
C
C     ****************************************************************
C     OUTPUT TO DATA FILE
C     ****************************************************************
      WRITE(8,103) NUMPTOUT, NUMSETS + 1
      WRITE(8,102) RADMIN, RADMAX
      DO 400 N = 1,NUMPTOUT
      WRITE(8,101) N,(RADIUS(N,J), J=1,NUMSETS)
  400 CONTINUE
C
C     ****************************************************************
      CLOSE(7)
      CLOSE(8)
  100 FORMAT(A)
  101 FORMAT(I5,5(E14.6))
  102 FORMAT(2(E14.6))
  103 FORMAT(2(I5))
  110 FORMAT(//////////////////////5X,'WELCOME TO EMCADIN',//5X,
     1'THIS PROGRAM ACCEPTS PROPERLY FORMATTED INPUT DATA',/5X,
     2'FROM  CURVE DIGITIZER AND CONVERTS IT TO A FORM.',/5X,
     3'WHICH CAN BE USED BY EMCAD. ',/5X,
     4////5X,'PLEASE PRESS ENTER TO CONTINUE.',//5X)
  120 FORMAT(//////////////////////5X,'INPUT FILENAME',//////5X,
     1'INPUTFN IS THE INPUT FILENAME OF THE FILE CONTAINING',/5X,
     2'THE DATA OF INTEREST WHICH WILL BE INTERPOLATED ON AND',/5X,
     3'CONVERTED TO A FORM WHICH CAN SUBSEQUENTLY BE USED BY',/5X,
     4'EMCAD.',///5X,
     5'PLEASE ENTER THE NAME OF THE INPUT FILE.  THE EXTENSION',/5X,
     6'OF THE FILENAME MUST BE .DAT, I.E. FILENAME.DAT',//5X)
  130 FORMAT(/////////////////////5X,'OUTPUT FILENAME',//////5X,
     1'OUTPUTFN IS THE OUTPUT FILENAME OF THE FILE CONTAINING',/5X,
     2'THE DATA WHICH HAS BEEN INTERPOLATED AND CONVERTED TO A',/5X,
```

```
   3'FORM WHICH WILL BE USED BY EMCAD.',///5X,
   4'PLEASE ENTER THE NAME OF THE OUTPUT FILE.  THE EXTENSION',/5X,
   6'OF THE FILENAME MUST BE INCLUDED I.E. FILENAME.DAT',//5X)
140 FORMAT(/////////////////////5X,'RESOLUTION',//////5X,
   1'DELTHE IS THE USER INPUT VALUE OF THE DESIRED THETA',/5X,
   2'RESOLUTION IN DEGREES.',///5X,
   3'PLEASE ENTER THE DESIRED DELTA THETA VALUE IN DEGREES.',//5X)
    END
```

APPENDIX F

CRVDIGIN USER'S MANUAL

Welcome to CRVDIGIN. This program is designed to accept properly formatted input data, and to convert it to a form which can be used by the West Coast Consultants commercial shareware product CURVE DIGITIZER (Appendix M). CURVE DIGITIZER is a computer aided design (CAD) package which will allow user manipulation of the data. Upon completion of using CURVE DIGITIZER, the user may employ the program EMCADIN to transform the data into a form which can be used by the scattering calculation program EMCAD.

A. BACKGROUND

EMSCAT was written by Dr. M. A. Morgan while at the University of California at Berkeley. Extensive interactive computer aided design type feature have been added to EMSCAT, which consider scattering from inhomogeneous axisymmetric objects. These computer aided design features, driving the change of program name from EMSCAT to EMCAD, allow structures composed of multiple dielectric materials to be graphically constructed or modified by the user, and then evaluated by EMCAD for the scattering. CRVDIGIN was written in the programming language of Ryan-McFarland FORTRAN (Appendix M). Any questions concerning formats, changes to the program, etc. should be referred to the

respective user's manuals, Dr. M. A. Morgan or LT. E. M. Connolly (Appendix M).

B. COMPUTER SETUP

CRVDIGIN was not written for any specific computer configuration, and as such, should run on most IBM compatible machines.

C. PROGRAM INPUT AND OUTPUT

CRVDIGIN obtains its input from a previously generated data file. The input data file for CRVDIGIN must be located in the same directory/subdirectory as CRVDIGIN. This data file may be produced by one of two methods. The first method consists of special case computer programs which generate data files describing the penetrable body of revolution for the specific geometrically shaped object being considered. A few examples of these special case programs are SPHERE, CONE and CYLINDER listed in Appendices J, K and L respectively. Each of these programs generates two descriptive data files which can be separated into distinct areas of information. The first area of information is the data which conveys the material composition of each layer of the structure. This data is of no concern at this particular time, however, it will be required before running EMCAD. The second area of information, which is the data required as input for CRVDIGIN, is the data which defines the structural shape of

each layer of the penetrable body. This data is referred to as "STRUCTUR".DAT, and is shown in the flow diagram of Figure F-1. The second method of input data file construction consists of user generation through direct keyboard input.

As shown in Figure F-1, the output of CRVDIGIN, shown as "CURVEIN".DAT, is the input to the CAD package CURVE DIGITIZER. For CRVDIGIN and CURVE DIGITIZER to be able to accept input data from a variety of sources, it was necessary to establish and adhere to a predefined data format. The special case computer programs were written to incorporate this format. In order for the second method to be utilized successfully, as well as for other special case programs to be written, it is imperative that the user understand the need for the particular format being used, as well as the format itself. The input format for CRVDIGIN is relatively uncomplicated. An example of a portion of an input data file for CRVDIGIN is shown in Figure F-2. The first line of data contains two numbers in integer format. The first integer represents the number of data points used to describe any layer. The second integer lists the total number of layers + 1. The '+ 1' was used to correct some minor discrepancies during the design of the code. The data defining the physical structure of the boundary for each layer is written in radius versus theta format. The value of the angle theta associated with each radius is determined

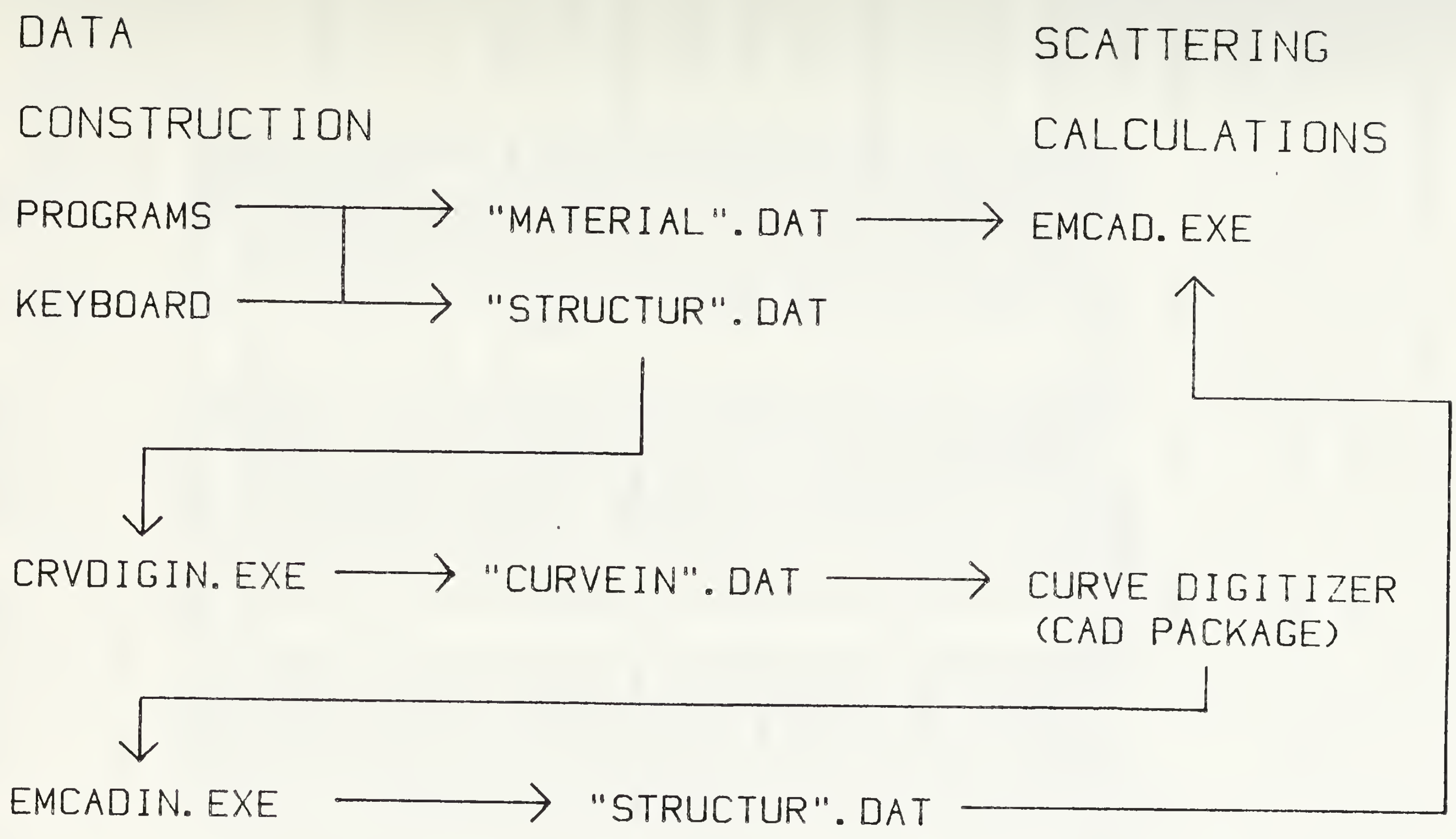


FIG. F-1. CRVDIGIN AND EMCADIN FLOW DIAGRAM

```
361    4
0.250300E+01  0.382760E+02
  1  0.188496E+02  0.251327E+02  0.314159E+02
  2  0.176975E+02  0.241175E+02  0.304952E+02
  3  0.166793E+02  0.231827E+02  0.296292E+02
  4  0.157731E+02  0.223194E+02  0.288130E+02
  5  0.149613E+02  0.215196E+02  0.280428E+02
  6  0.142301E+02  0.207767E+02  0.273146E+02
  7  0.135680E+02  0.200848E+02  0.266253E+02
  8  0.129657E+02  0.194390E+02  0.259719E+02
  9  0.124155E+02  0.188348E+02  0.253516E+02
 10  0.119109E+02  0.182684E+02  0.247621E+02
 11  0.114466E+02  0.177363E+02  0.242013E+02
 12  0.110180E+02  0.172357E+02  0.236670E+02
 13  0.106211E+02  0.167638E+02  0.231575E+02
 14  0.102525E+02  0.163182E+02  0.226712E+02
 15  0.990939E+01  0.158969E+02  0.222066E+02
 16  0.958821E+01  0.154980E+02  0.217622E+02
 17  0.928975E+01  0.151197E+02  0.213369E+02
 18  0.900810E+01  0.147605E+02  0.208294E+02
 19  0.874555E+01  0.144191E+02  0.205387E+02
 20  0.849761E+01  0.140941E+02  0.201639E+02
 21  0.826396E+01  0.137845E+02  0.198040E+02
 22  0.804341E+01  0.134893E+02  0.194582E+02
 23  0.783490E+01  0.132074E+02  0.191256E+02
 24  0.763750E+01  0.129380E+02  0.188057E+02
 25  0.745035E+01  0.126803E+02  0.184976E+02
 26  0.727270E+01  0.124336E+02  0.182009E+02
 27  0.710385E+01  0.121972E+02  0.179149E+02
 28  0.694318E+01  0.119706E+02  0.176390E+02
 29  0.679012E+01  0.117531E+02  0.173728E+02
 30  0.664416E+01  0.115442E+02  0.171158E+02
```

FIG. F-2. PORTION OF SAMPLE INPUT DATA FOR CRVDIGIN.EXE

by starting at theta equal to 0 degrees and progressing to theta equals 180 degrees in step sizes determined by the number of points available. The program expects that the values of the radius for each layer have been entered starting with the layer closest to the center of the object, then advancing outward from the center until all layers have been accounted for. The second line of data contains two real values. The first real value is the minimum radius value present, and the second value is the maximum radius value present. Following the description for layer entry above, this implies that the minimum radius value should come from the first layer, and the maximum radius value should come from the last layer. Figure F-3 demonstrates the indexing of layers. The next A lines of data consists of B columns of data. The value of A is the number of points, and the value of B is the number of layers + 1. The first column value is an index value starting at 1 and incrementing by 1. The second, third, ... columns contain values of the radius associated with theta = (index - 1) * deltatheta for the first layer, the second layer ... For each successive row the index changes, the value of theta changes and the radii associated with that given value of theta are listed.

The output of CRVDIGIN, used as an input to CURVE DIGITIZER, is just as simple in appearance as the input to CRVDIGIN, yet much more difficult in theory. The

MULTI-LAYERED, MULTI-HEIGHT
MULTI-RADIUS CYLINDER

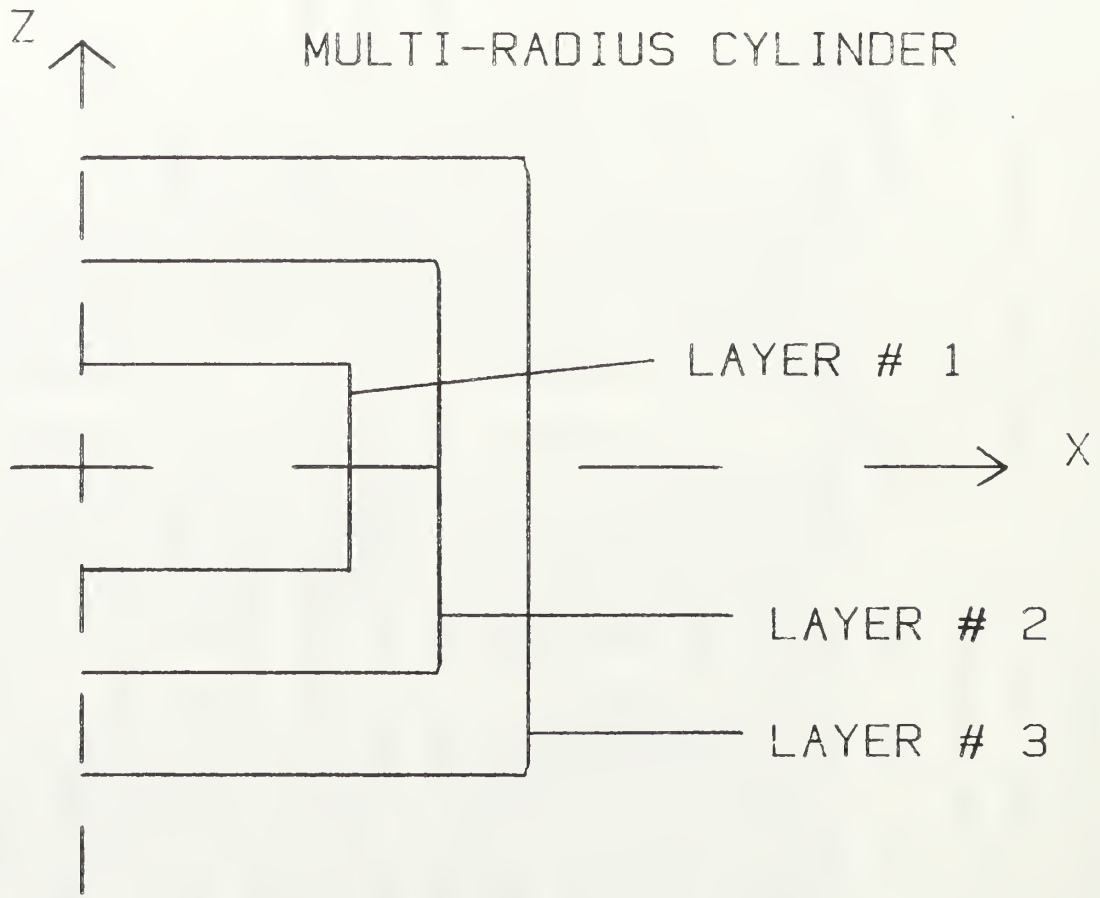


FIG. F-3. GRAPHICAL REPRESENTATION OF ASSIGNMENT OF LAYER BOUNDARIES

explanation of this difficulty actually is embedded in the operation of CURVE DIGITIZER. Although these comments may apply to other CAD packages, they are specifically directed towards CURVE DIGITIZER, referred to henceforth for brevity as CD. Computer aided graphic design deals in the basic business of handling points. Any graphic figure can be represented as some combination of points. Lines can be formed by connecting two points. Arcs can be represented by placing more points closer together and by approximating the curvature with sufficiently small line segments between points. The required input-output format for CD is utilized to insure that adequate information is present to direct CD how to connect the points. A similar process may be that of the simple children's game connect-the-dots. A series of dots exists which define a picture, if connected properly. As more and more dots are used, more complex pictures with better resolution can be constructed. The fundamental assumption employed by CD is that each point should be connected to the next point listed in the data file unless some code is present to indicate otherwise. For simplicity and unrestricted accessibility, CD input and output data are represented in terms of XY rectangular coordinates. CD flags the break between two data points by what shall be referred to as a separation code. Since two coordinates, X and Y, are required to establish a data point, two separation codes, one for the X data and one for

the Y data are required to separate two distinct data points. The separation code employed is of the form 999990. The last two digits may change to reflect scaling, color, or any of a number of other things. It was not possible to determine the specifics from the user manual. In the specific case of CRVDIGIN output data, 999990 999990 codes are used to distinguish the structural data describing one layer from that describing another layer. Other than the insertion of separation codes at the appropriate locations, CRVDIGIN performs the simple mathematical task of converting the input data generated in radius-theta format to output data represented in XY format.

D. PROGRAM EXECUTION

Once the correct disk or directory has been selected, CRVDIGIN can be invoked by typing

CRVDIGIN

at the DOS prompt. The program will load itself and the user will see the following screens:

* Welcome to CRVDIGIN (Figure F-4)

This screen presents some brief background information.

```
WELCOME TO CRVDIGIN

THIS PROGRAM ACCEPTS PROPERLY FORMATTED INPUT DATA
AND CONVERTS IT TO A FORM WHICH CAN BE USED BY THE
COMMERCIAL SHAREWARE PRODUCT CURVE DIGITIZER.
CURVE DIGITIZER IS A CAD PACKAGE WHICH WILL ALLOW USER
MANIPULATION OF THE DATA.  UPON COMPLETION OF
CURVE DIGITIZER, THE USER MAY UTILIZE THE PROGRAM
EMCADIN TO TRANSFORM THE DATA INTO A FORM WHICH CAN
BE USED BY THE PROGRAM EMCAD WRITTEN BY DR.  M. A. MORGAN.


PLEASE PRESS ENTER TO CONTINUE.
```

FIG. F-4. WELCOME TO CRVDIGIN MENU

* Input Filename (Figure F-5)

This screen prompts the user for the name of the file containing the input data for CRVDIGIN. The file may have any extension that the user desires, however, the extension must be entered by the user as part of the filename. The input data file must be in the same directory as CRVDIGIN.

* Output Filename (Figure F-6)

This screen prompts the user for the name of the file that will hold the output data from CRVDIGIN. Because of the input requirements of CURVE DIGITIZER, the extension of this file must be .dat, and must be included as part of the output filename.

* Resolution (Figure F-7)

CRVDIGIN uses linear interpolation to provide the capability to generate output data with better resolution than the input data. This screen prompts the user for the desired output data resolution, in the units of degrees.

E. PROGRAM OUTPUT

A portion of an example of CRVDIGIN output is shown in Figure F-8. Notice the 999990 999990 codes separating the different layers of data.

INPUT FILENAME

INPUTFN IS THE INPUT FILENAME OF THE FILE CONTAINING THE DATA OF INTEREST WHICH WILL BE INTERPOLATED ON AND CONVERTED TO A FORM WHICH CAN SUBSEQUENTLY BE USED BY THE CURVE DIGITIZER PROGRAM.

PLEASE ENTER THE NAME OF THE INPUT FILE. THE EXTENSION OF THE FILENAME MUST BE INCLUDED, I.E. FILENAME.DAT

FIG. F-5. CRVDIGIN PROMPT FOR INPUT FILENAME

OUTPUT FILENAME

OUTPUTFN IS THE OUTPUT FILENAME OF THE FILE CONTAINING THE DATA WHICH HAS BEEN INTERPOLATED AND CONVERTED TO A FORM WHICH WILL BE USED BY THE CURVE DIGITIZER PROGRAM.

PLEASE ENTER THE NAME OF THE OUTPUT FILE. THE EXTENSION OF THE FILENAME MUST BE .DAT, I.E. FILENAME.DAT

FIG. F-6. CRVDIGIN PROMPT FOR OUTPUT FILENAME

RESOLUTION

DELTHE IS THE USER INPUT VALUE OF THE DESIRED THETA
RESOLUTION IN DEGREES.

PLEASE ENTER THE DESIRED DELTA THETA VALUE IN DEGREES.

FIG. F-7. CRVDIGIN PROMPT FOR OUTPUT DATA RESOLUTION

4.69975948	-18.84953690
4.35180950	-18.84956930
4.00664711	-18.84958080
3.66403151	-18.84959030
3.32372093	-18.84950830
2.98551798	-18.84953120
2.64917397	-18.84954830
2.31447673	-18.84953880
1.98121083	-18.84956740
1.64916909	-18.84959410
1.31813526	-18.84956740
0.98790431	-18.84953120
0.65828347	-18.84950830
0.32906568	-18.84952740
4.32930647E-05	-18.84959980
999990	999990
0.00000000E-01	25.13270000
0.40459359	23.17916870
0.75102264	21.50649070
1.05115640	20.05727580
1.31384826	18.78891940
1.54581928	17.66880800
1.75229287	16.67196660
1.93734324	15.77840710
2.10425401	14.97255710
2.25564241	14.24157710
2.39365172	13.57508280

FIG. F-8. PORTION OF SAMPLE OUTPUT DATA FROM CRVDIGIN

APPENDIX G

CRVDIGIN SOURCE CODE

```
C       PROGRAM CRVDIGIN
C
C       *****************************************************************
C
C       WELCOME TO CRVDIGIN WRITTEN BY LT  E. M. CONNOLLY.
C       CRVDIGIN IS A PROGRAM DESIGNED TO ACCEPT PROPERLY FORMATTED
C       INPUT DATA, AND CONVERT IT TO A FORM WHICH CAN BE USED BY
C       THE COMMERCIAL SHAREWARE PRODUCT CURVE DIGITIZER.
C       CURVE DIGITIZER IS A CAD PACKAGE WHICH WILL ALLOW USER
C       MANIPULATION OF THE DATA.  UPON COMPLETION OF CURVE DIGITIZER,
C       THE USER MAY UTILIZE THE PROGRAM EMCADIN, ALSO WRITTEN BY
C       LT  E. M. CONNOLLY TO TRANSFORM THE DATA INTO A FORM WHICH CAN
C       BE USED BY THE PROGRAM EMCAD WRITTEN BY DR.  M. A. MORGAN.
C
C       *****************************************************************
C       VARIABLE DEFINITIONS
C       *****************************************************************
C
C       INPUTFN            THE INPUT FILENAME OF THE FILE CONTAINING
C                          THE DATA OF INTEREST WHICH WILL BE INTERPOLATED
C                          ON AND CONVERTED TO A FORM WHICH CAN
C                          SUBSEQUENTLY BE USED BY THE CURVE DIGITIZER
C                          PROGRAM
C       OUTPUTFN           THE OUTPUT FILENAME OF THE FILE CONTAINING
C                          THE DATA WHICH HAS BEEN INTERPOLATED AND
C                          CONVERTED TO A FORM WHICH WILL BE USED BY
C                          THE CURVE DIGITIZER PROGRAM
C       I, J, K,           INTEGER COUNTERS
C       L, N
C       NUMBROWS           THE NUMBER OF ROWS CONTAINED IN THE DATA
C                          MATRIX IN THE FILE INPUTFN
C       NUMBCOLS           THE NUMBER OF COLUMNS CONTAINED IN THE DATA
C                          MATRIX IN THE FILE INPUTFN
C       SEPRATN            A FLAG USED BY CURVE DIGITIZER TO DISTINGUISH
C                          BETWEEN DATA POINTS OR SETS OF DATA
C       RADARRAY           A WORKING ARRAY USED TO HOLD ALL THE VALUES
C                          OF THE ARRAY SPECIFIED ABOVE BY NUMBROWS AND
C                          NUMBCOLS
C       MINRAD             THE MINIMUM RADIUS VALUE, NOT USED BY THIS
C                          PROGRAM BUT REQUIRED FOR EMCAD
C       MAXRAD             THE MAXIMUM RADIUS VALUE, NOT USED BY THIS
C                          PROGRAM BUT REQUIRED FOR EMCAD
C       RADIUS             A WORKING VARIABLE USED TO HOLD THE CURRENT
C                          VALUE OF THE RADIUS OBTAINED FROM RADARRAY
C       PI                 A CONSTANT
C       DELTHE             THE USER INPUT VALUE OF THE DESIRED THETA
C                          RESOLUTION IN DEGREES
C       DELTHETA           THE USER INPUT VALUE OF THE DESIRED THETA
C                          RESOLUTION IN RADIANS
C       THETAOLD           A WORKING VARIABLE USED TO HOLD THE CURRENT
C                          VALUE OF THETA OBTAINED FROM RADARRAY
C       THETA              AN ARRAY HOLDING THE DESIRED VALUES OF THETA
C                          AS DETERMINED BY DELTHE
C       DEGTORAD           A CONVERSION FACTOR FROM DEGREES TO RADIANS
C       X                  THE VALUE OF X OBTAINED BY CONVERTING THE POLAR
C                          COORDINATES RADIUS,THETA TO X,Y COORDINATES
C                          THE X,Y COORDINATE SYSTEM IS REQUIRED FOR THE
C                          DATA TO BE COMPATIBLE WITH THE CURVE DIGITIZER
```

```
C                             PROGRAM
C     Y                       THE VALUE OF Y OBTAINED BY CONVERTING THE POLAR
C                             COORDINATES RADIUS,THETA TO X,Y COORDINATES
C                             THE X,Y COORDINATE SYSTEM IS REQUIRED FOR THE
C                             DATA TO BE COMPATIBLE WITH THE CURVE DIGITIZER
C                             PROGRAM
C     INTER1, INTER2          INTERMEDIATE CALCULATION VARIABLES
C
C     ******************************************************************
C     VARIABLE DECLARATION
C     ******************************************************************
      CHARACTER*12 INPUTFN, OUTPUTFN, DUMMY
      INTEGER I, J, K, L, N
      INTEGER NUMBROWS, NUMBCOLS, NUMPTS, SEPRATN
      REAL RADARRAY(1000,10)
      REAL MINRAD, MAXRAD, RADIUS, PI, DELTHE, DELTHETA, THETAOLD
      REAL THETA(1000), DEGTORAD, X, Y
      REAL INTER1, INTER2
C
C     ******************************************************************
C     INITIAL VALUES
C     ******************************************************************
      PI = 3.14159
      DEGTORAD = PI / 180.
      SEPRATN = 999990
C
C     ******************************************************************
C     INPUT DATA OF INTEREST
C     ******************************************************************
      WRITE(*,110)
      READ(*,100) DUMMY
      WRITE(*,120)
      READ(*,100) INPUTFN
      WRITE(*,130)
      READ(*,100) OUTPUTFN
      WRITE(*,140)
      READ(*,*) DELTHE
C
C     ******************************************************************
C     READ DATA FILE
C     ******************************************************************
      OPEN(7,FILE = INPUTFN)
      OPEN(8,FILE = OUTPUTFN)
      READ(7,*) NUMBROWS, NUMBCOLS
      READ(7,*) MINRAD, MAXRAD
      READ(7,*) ((RADARRAY(I,J), J=1,NUMBCOLS), I = 1,NUMBROWS)
C
C     ******************************************************************
C     INTERMEDIATE CALCULATIONS
C     ******************************************************************
      NUMPTS = INT(180. / DELTHE) + 1
      DELTAOLD = (180. / (NUMBROWS - 1)) * DEGTORAD
      DELTHETA = DELTHE * DEGTORAD
C
C     ******************************************************************
C     INTERPOLATION AND TRANSFORMATION
C     ******************************************************************
      J = 1
      DO 10 K = 2, NUMBCOLS
      DO 20 N = 1, NUMPTS
      THETA(N) = (N-1) * DELTHETA
      DO 30 I = 1, NUMBROWS
      THETAOLD = (RADARRAY(I,J) - 1) * DELTAOLD
      IF(ABS(THETA(N) - THETAOLD) .LT. .00001) THEN
      RADIUS = RADARRAY(I,K)
      X = RADIUS * SIN( THETA(N) )
```

```
      Y = RADIUS * COS( THETA(N) )
      WRITE(8,*) X,Y
      GOTO 20
      ELSE IF (THETA(N) .GE. (THETAOLD - DELTAOLD) .AND.
     1       THETA(N) .LE. THETAOLD) THEN
      L = I - 1
      INTER1 = RADARRAY(L,K) - RADARRAY(I,K)
      INTER2 = (THETA(N) - THETAOLD) / (-DELTAOLD)
      RADIUS = INTER1 * INTER2 + RADARRAY(I,K)
C     REMOVES NORMALIZING FACTOR OF 2*PI GENERATED BY SPHERE, CONE,
C     ETC.., FACTOR IS RESTORED IN EMCADIN
      X = RADIUS * SIN( THETA(N) ) * 1 / ( 2 * PI )
      Y = RADIUS * COS( THETA(N) ) * 1 / ( 2 * PI )
      WRITE(8,*) X,Y
      GOTO 20
      ELSE
C     DUMMY BRANCH, TRY NEXT VALUE OF THETAOLD
      RADIUS = 0
      ENDIF
   30 CONTINUE
   20 CONTINUE
      WRITE(8,*) SEPRATN,SEPRATN
   10 CONTINUE
C
C     *****************************************************************
  100 FORMAT(A)
  110 FORMAT(////////////////////5X,'WELCOME TO CRVDIGIN',//5X,
     1'THIS PROGRAM ACCEPTS PROPERLY FORMATTED INPUT DATA',/5X,
     2'AND CONVERTS IT TO A FORM WHICH CAN BE USED BY THE ',/5X,
     3'COMMERCIAL SHAREWARE PRODUCT CURVE DIGITIZER.',/5X,
     4'CURVE DIGITIZER IS A CAD PACKAGE WHICH WILL ALLOW USER',/5X,
     5'MANIPULATION OF THE DATA.  UPON COMPLETION OF',/5X,
     6'CURVE DIGITIZER, THE USER MAY UTILIZE THE PROGRAM ',/5X,
     7'EMCADIN TO TRANSFORM THE DATA INTO A FORM WHICH CAN',/5X,
     8'BE USED BY THE PROGRAM EMCAD WRITTEN BY DR.  M. A. MORGAN.',
     9////5X,'PLEASE PRESS ENTER TO CONTINUE.',//5X)
  120 FORMAT(////////////////////5X,'INPUT FILENAME',//////5X,
     1'INPUTFN IS THE INPUT FILENAME OF THE FILE CONTAINING',/5X,
     2'THE DATA OF INTEREST WHICH WILL BE INTERPOLATED ON AND',/5X,
     3'CONVERTED TO A FORM WHICH CAN SUBSEQUENTLY BE USED BY',/5X,
     4'THE CURVE DIGITIZER PROGRAM.',///5X,
     5'PLEASE ENTER THE NAME OF THE INPUT FILE.  THE EXTENSION',/5X,
     6'OF THE FILENAME MUST BE INCLUDED, I.E. FILENAME.DAT',//5X)
  130 FORMAT(////////////////////5X,'OUTPUT FILENAME',//////5X,
     1'OUTPUTFN IS THE OUTPUT FILENAME OF THE FILE CONTAINING',/5X,
     2'THE DATA WHICH HAS BEEN INTERPOLATED AND CONVERTED TO A',/5X,
     3'FORM WHICH WILL BE USED BY THE CURVE DIGITIZER PROGRAM.',///5X,
     4'PLEASE ENTER THE NAME OF THE OUTPUT FILE.  THE EXTENSION',/5X,
     6'OF THE FILENAME MUST BE .DAT, I.E. FILENAME.DAT',//5X)
  140 FORMAT(////////////////////5X,'RESOLUTION',//////5X,
     1'DELTHE IS THE USER INPUT VALUE OF THE DESIRED THETA',/5X,
     2'RESOLUTION IN DEGREES.',///5X,
     3'PLEASE ENTER THE DESIRED DELTA THETA VALUE IN DEGREES.',//5X)
      CLOSE(7)
      CLOSE(8)
      END
```

APPENDIX H

MCADGRAF USER'S MANUAL

Welcome to MCADGRAF. This program was written to be used in conjunction with the FORTRAN program entitled EMCAD. MCADGRAF adds graphics features to support the analysis of data generated and formatted by EMCAD.

A. BACKGROUND

MCADGRAF was developed and written by LT. E. M. Connolly using the programming language of Ryan-McFarland FORTRAN (Appendix M). Microcompatibles GRAFMATIC (Appendix M), and later Jewell Technologies GRAFPLUS (Appendix M) were chosen as the commercially available screen graphics and hard copy support for EMCAD output graphics. Any questions concerning colors, formats, changes to the program, etc. should be referred to the respective user's manuals, Dr. M. A. Morgan or LT. E. M. Connolly (Appendix M).

B. COMPUTER SETUP

MCADGRAF was written for the following computer configuration:

* IBM AT or compatible, including 80386 based compatibles
* hard disk
* EGA graphics card

* EGA monitor
* math coprocessor - not required
* memory - 512K RAM minimum

MCADGRAF can be placed in any directory, at the user's discretion, however, it is highly recommended that MCADGRAF and EMCAD be placed in the same directory. MCADGRAF input is taken directly from EMCAD output. This output is required to be in the same directory/sub-directory as MCADGRAF, or on the same floppy disk if using a floppy system instead of a hard disk. The only other requirement for initial setup deals with printer initialization. In order to obtain a printer graphical output, GRAFPLUS must be installed on the computer, configured for the specific printer of use, prior to running MCADGRAF. An EPSON LX-800 series printer was used when writing this code. This printer, as with any other printer to be used, was configured by inserting two lines in the AUTOEXEC.BAT file. The first of these lines was a path command to tell the computer where to find the appropriate files for GRAFPLUS. Assuming all the files for GRAFPLUS are contained in a directory on the C disk entitled Grafplus, an appropriate path entry would be

```
path c:\;c:\Grafplus
```

This setup will also work with other IBM graphics compatible printers such as the Okidata ML - 192/193. The second of these lines inserted actually established the initialization values for the printer to be used. The second line must be inserted anywhere in the AUTOEXEC.BAT after the path command. The printer initialization used for the LX-800 was

GRAFPLUS = #I15

The equals sign (=) directs the printer to print the graphics horizontally. This means that the top of the image is at the top left corner of the page. For vertical printing, the object appears to be rotated with the top of the picture at the bottom left corner of the page. The pound sign (#) directs the printer to print the text screens as graphics screens. The I conveys the information that the screen mode is EGA - 640x350 resolution, with 16 colors available. The number 15 is the number assigned by GRAFPLUS to correspond with the EPSON LX-800. Other options available are the capabilities to print in color and to reverse printing. All of these options are discussed in much greater detail on pages 3 through 8 of the GRAFPLUS user's manual. As a note of caution, some GRAFPLUS printer definitions are defined such that "graying" is used on a black and white print-out to represent different colors. On the LX-800, the "graying" effect reduced printer clarity and

resolution dramatically. This problem was rectified by running the EDITGRAF.COM program contained in GRAFPLUS, answering all the questions without changes until the question "Do you wish to print in Black & White only? (disable gray scale) " appeared on the screen. Changing the answer from no to yes removed the "graying" and improved the resolution. A much more detailed explanation of EDITGRAF can be found on pages 9 through 21 of the GRAFPLUS user's manual. Once these changes to the printer definitions and AUTOEXEC.BAT have been made, it is necessary to re-boot the computer to initialize these values. This re-boot can be accomplished by either turning the machine off, waiting 15 to 20 seconds, and turning it back on, or by hitting the Ctrl, Alt and Del keys all at the same time.

C. PROGRAM INPUT

As previously mentioned, MCADGRAF obtains its input from the output of EMCAD. This EMCAD output is four files in the form of "FILENAME".TET, "FILENAME".TEP, "FILENAME".TMT and "FILENAME".TMP. The quotation marks indicate that the FILENAME is a value that was previously chosen by the user while running EMCAD. The four extensions of FILENAME have significant meaning. The TET refers to TE incidence, F-Theta; the TEP refers to TE incidence, F-Phi; the TMT refers to TM incidence, F-Theta; the TMP refers to TM incidence, F-Phi. The values for these four files were taken directly from a much larger, more comprehensive EMCAD

output data file. The scattering calculation output and graphics relationship is depicted in Figure H-1.

D. PROGRAM EXECUTION

MCADGRAF is a menu driven output graphics program. Once the correct disk or directory has been selected, MCADGRAF can be invoked by typing

MCADGRAF

at the DOS prompt. The program will load itself, and the user will see the following screens:

* Welcome To MCADGRAF (Figure H-2)

This screen presents some brief background information.

* EMCAD Output Graphics Presentation, Part 1 (Figure H-3)

This menu affords the user the opportunity to exit the program in the event that the printer initialization has not been accomplished.

* EMCAD Output Graphics Presentation, Part 2 (Figure H-4)

This screen explains the means of obtaining an output graph on the printer. The user will be presented with four graphs

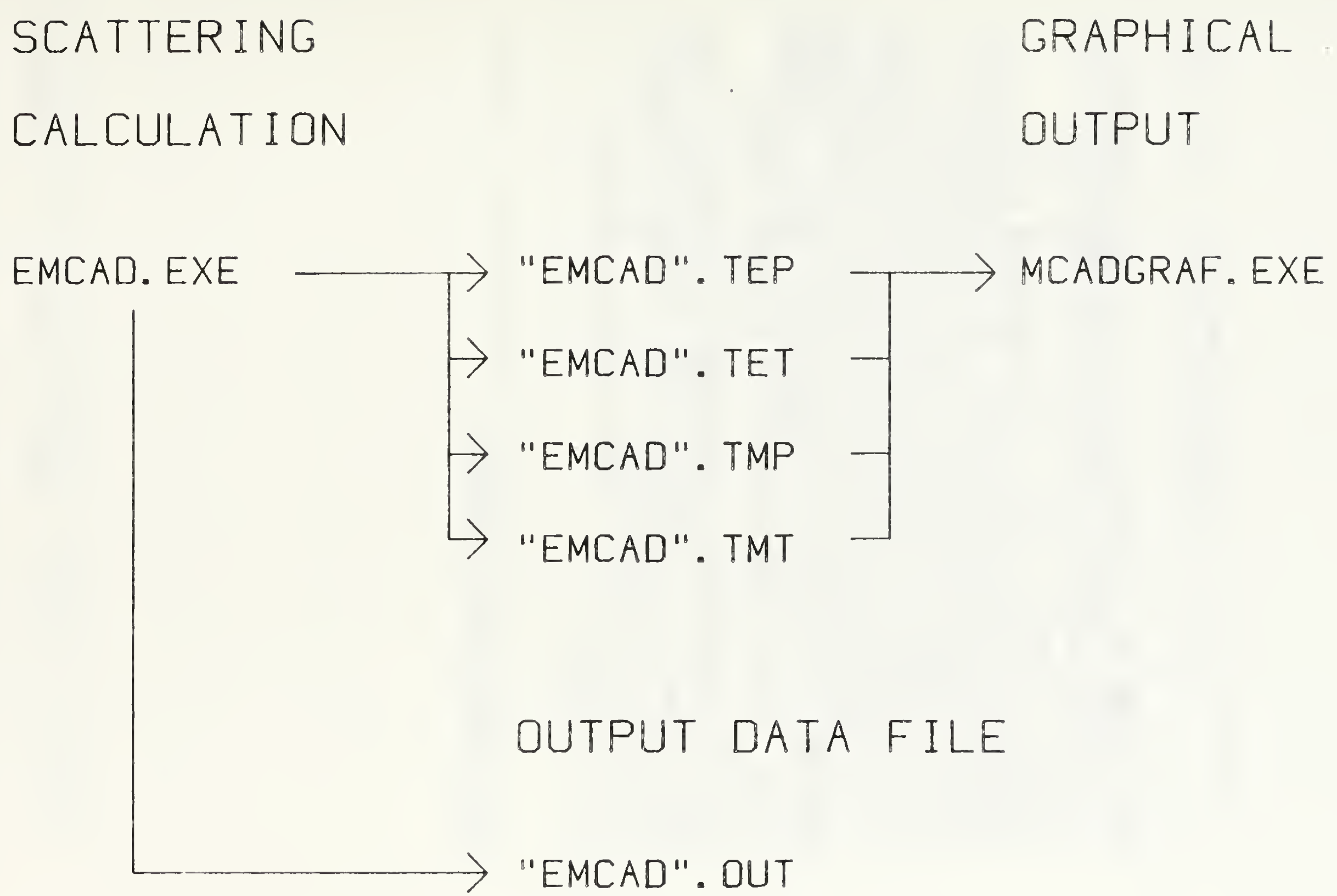


FIG. H-1. SCATTERING CALCULATION OUTPUT AND GRAPHICS

********** WELCOME TO MCADGRAF **********

THIS PROGRAM WAS WRITTEN TO BE USED IN CONJUNCTION WITH
THE FORTRAN PROGRAM ENTITLED EMCAD. MCADGRAF ADDS
GRAPHING FEATURES TO SUPPORT THE ANALYSIS OF DATA
GENERATED AND FORMATTED BY EMCAD. EMCAD WAS WRITTEN BY
DR. M. A. MORGAN, WITH MODIFICATIONS BY HIMSELF AND
LT E. M. CONNOLLY. MCADGRAF WAS DEVELOPED AND WRITTEN BY
LT E. M. CONNOLLY TO HELP FULFILL THESIS REQUIREMENTS WHILE
UNDER THE GUIDANCE AND THE SUPPORT OF DR. MORGAN. MCADGRAF
WAS WRITTEN AND COMPILED USING RYAN MCFARLAND FORTRAN,
GRAFMATIC, AND GRAFPLUS. ANY QUESTIONS CONCERNING COLORS,
FORMAT, CHANGES TO THE PROGRAM, ETC... SHOULD BE REFERRED
TO THE RESPECTIVE USERS MANUALS, DR. M. A. MORGAN, OR
LT E. M. CONNOLLY.

PLEASE PRESS ENTER TO CONTINUE.

FIG. H-2. WELCOME TO MCADGRAF

EMCAD OUTPUT GRAPHICS PRESENTATION

NOTES CONCERNING GRAPHICAL HARDCOPY OUTPUT, PART 1

This program allows the user to obtain printer hardcopies of the graphs being viewed. Hardcopy graphic outputs are obtained through the use of the program entitled GRAFPLUS. Proper utilization of MCADGRAF requires prior setup of GRAFPLUS. Directions for proper setup are contained in the GRAFPLUS users manual.

1. CONTINUE WITH THE NEXT SCREEN

2. EXIT THE PROGRAM

Indicate your selection by the choice of 1 or 2.

FIG. H-3. EMCAD OUTPUT GRAPHICS PRESENTATION PART 1

EMCAD OUTPUT GRAPHICS PRESENTATION

NOTES CONCERNING GRAPHICAL HARDCOPY OUTPUT, PART 2

The user makes the decision as to whether a printer graphical output is desired after viewing the graph. Once a graph is drawn on the screen, the user may request a printer hardcopy by typing either a "P" or "p" and pressing the "ENTER" key. Selection of any other key will simply advance to the next graph. Spacing on 8 1/2 by 11 paper allows up to 2 graphs per page.

PLEASE PRESS ENTER TO CONTINUE.

FIG. H-4. EMCAD OUTPUT GRAPHICS PRESENTATION PART 2

for each data run. As each graph is presented on the screen, the user is given the opportunity to obtain a printer copy of the screen. The user decides to obtain a printer copy by depressing a P or a p, followed by ENTER. The reminder line telling the user what to do to produce a hardcopy is erased, and the graph is sent to the printer. Any other character input, followed by ENTER will simply advance to the next screen. Setting up the paper in the printer is the sole responsibility of the user for each and every graph. If one graph is to placed on every page, it is recommended that the paper be advanced a few inches so that the graph is printed closer to the center of the page. For the LX-800, it is possible to place 2 graphs on every page, if the paper is positioned properly in the printer. The best advice is to experiment.

* Data File Name Input Menu (Figure H-5)

If the data file to be used was generated by EMCAD, the user need only choose option 1. If the file to be used is a self-generated file, it must be in the exact format of an EMCAD generated file, and it must have an extension of .TET, .TEP, .TMT or .TMP . If the file is not in the correct format, or if the file does not contain the proper extension, the user must exit the program and correct the problem prior to continuing.

EMCAD OUTPUT GRAPHICS PRESENTATION

DATA FILE NAME INPUT MENU

If utilizing a file or series of files created by EMCAD, please select option 1. If utilizing any file other than one created by EMCAD, please ensure that the file is in proper format, and that the filename extension is one of the four extensions utilized by this program. These extensions are: .tet .tep .tmt and .tmp
If you are unsure as to whether your file is in the proper format with the proper extension, please exit the program and verify before continuing.

1. INPUT THE FILE OF USER CHOICE

2. EXIT THE PROGRAM

Indicate your selection by the choice of 1 or 2.

FIG. H-5. DATA FILE NAME INPUT MENU

* Data File Name Input (Figure H-6)

This menu prompts the user for the filename of the input data file. Since the extension is limited to four cases, and has previously been selected in the Data File Type Selection Menu, no extension should be added to the filename to be typed in.

* Data File Type Selection Menu (Figure H-7)

As previously discussed, MCADGRAF requires that the four EMCAD output data files be located in the same directory. This menu gives the user the chance to choose which of the four files the data will come from.

This menu also affords the user the opportunity to change the default setting of the aspect ratio. The menu for this change is shown in Figure H-8. Aspect ratio is a term that relates the number of available pixel columns on the screen to the number of available pixel rows on a screen. Since most computer screens are not square, this representation becomes very important. In simple terms, it deals with making a square look like a square, and not like a rectangle, or making a circle look like a circle, and not an oval.

GRAFPLUS comes with a program to help determine the aspect ratio for any monitor. In the case of the

EMCAD OUTPUT GRAPHICS PRESENTATION

DATA FILE NAME INPUT

Please input the name of the file containing data that you wish to work with. Please input only the file name i.e. do not include the file extension. The file extension will be selected in the DATA FILE TYPE SELECTION MENU.

FIG. H-6. DATA FILE NAME INPUT

EMCAD OUTPUT GRAPHICS PRESENTATION
DATA FILE TYPE SELECTION MENU

Please select the type of data file containing the information of interest. EMCAD automatically created these four data file type s.

1. TE INCIDENCE, F-THETA

2. TE INCIDENCE, F-PHI

3. TM INCIDENCE, F-THETA

4. TM INCIDENCE, F-PHI

5. CHANGE DATA FILE

6. CHANGE ASPECT RATIO

7. EXIT THE PROGRAM

Indicate your selection by the choice of 1, 2, 3, 4, 5, 6 or 7

FIG. H-7. DATA FILE TYPE SELECTION MENU

********** ASPECT RATIO **********

IN ORDER TO AVOID DISTORTION ON THE SCREEN, IT IS NECESSARY TO DEFINE THE NUMBER OF ROWS PER INCH AND THE NUMBER OF COLUMNS PER INCH. THESE DEFINITIONS CHANGE AS THE RESOLUTION OF THE SCREEN, AS WELL AS THE TYPE OF SCREEN CHANGE. THIS PROGRAM WAS WRITTEN IN MODE 16 (EGA) USING A GB-1 VIDEO BOARD AND A NEC MULTISYNC MONITOR. THE OPTIMUM ASPECT RATIO FOR THIS CONFIGURATION WAS .65

PLEASE INPUT ASPECT RATIO.

FIG. H-8. ASPECT RATIO

development of this program, the aspect ratio given by GRAFPLUS delivered erroneous results. As such, this screen of MCADGRAF allows the user to input and change the aspect ratio of choice. For users unsure of their aspect ratio, start with an initial value of 1, and work up or down with increments of .1 or .2 until close to the desired results, then fine tune as much as needed.

* Selection of Incidence Angle (Figure H-9)

One of the input parameters for EMCAD was the number of incident fields, and their respective angles. MCADGRAF reads the input data file, and presents in this menu the number of incidence angles, and their respective angles in degrees. It is up to the user to select the incidence angle of interest.

* Selection of Phi Angle (Figure H-10)

As with incidence angles, phi angles were input parameters for EMCAD. MCADGRAF presents in this menu the number of phi angles, and their respective angles in degrees. It is up to the user to select the phi angle of interest.

```
EMCAD OUTPUT GRAPHICS PRESENTATION

SELECTION OF INCIDENCE ANGLE
The data file you have selected contains data for  5
incidence angles.  Please select your choice from the
following menu.  Note, you may only work with one
incidence angle at a time.

1     INCIDENCE ANGLE =    0.00
2     INCIDENCE ANGLE =   45.00
3     INCIDENCE ANGLE =   90.00
4     INCIDENCE ANGLE =  135.00
5     INCIDENCE ANGLE =  180.00
6     RETURN TO THE DATA FILE TYPE SELECTION MENU
7     EXIT THE PROGRAM

INDICATE YOUR SELECTION BY THE CHOICE OF A NUMBER  I.E. 1  2  3 ...)
```

FIG. H-9. SELECTION OF INCIDENCE ANGLE

```
EMCAD OUTPUT GRAPHICS PRESENTATION

SELECTION OF PHI ANGLE
The data file you have selected contains data for  3
phi angles.  Please select your choice from the following
menu.  Note, you may only work with one phi angle at a time.

 1     PHI ANGLE =    0.00
 2     PHI ANGLE =   90.00
 3     PHI ANGLE =  180.00
 4     RETURN TO THE DATA FILE TYPE SELECTION MENU
 5     EXIT THE PROGRAM

INDICATE YOUR SELECTION BY THE CHOICE OF A NUMBER  I.E. 1  2  3 ...)
```

FIG. H-10. SELECTION OF PHI ANGLE

E. PROGRAM OUTPUT

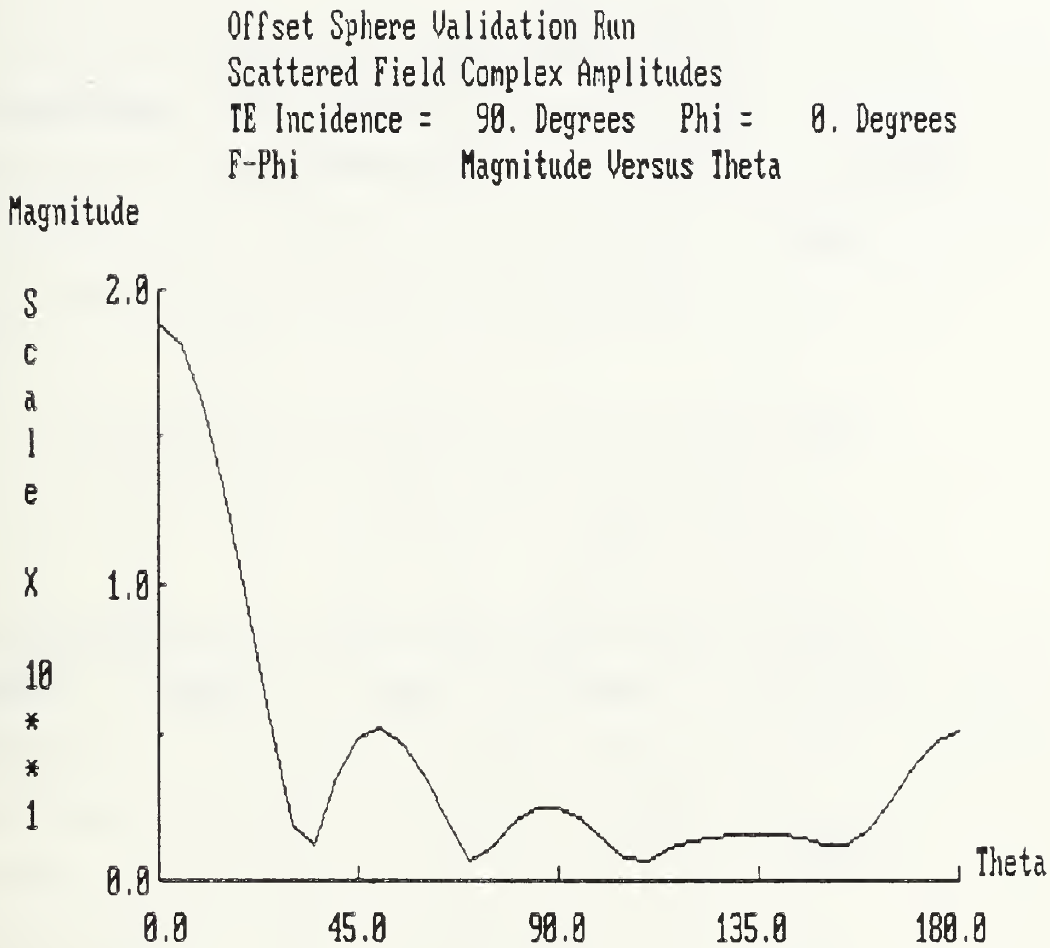
For each separate data set, MCADGRAF produces four different graphs.

* Magnitude Versus Theta (Figure H-11)

As can be seen in the first line of the title for this graph, as well as the other three graphs, a label specific to this data set appears. This label was used as an input parameter to EMCAD to assist the user in differentiating this set of data from every other set. In the third line, the values of the incidence angle and phi angle, as well as the type of incidence are listed. Either F-Phi or F-Theta appears in the fourth line followed by a short description of the type of graph. The line at the bottom referring to hardcopy is simply a reminder to the user of what is needed to request a hardcopy output. This line is erased before the printer graph is produced. This first graph depicts the scattering magnitude plotted versus the 180 degrees of Theta.

* Phase Versus Theta (Figure H-12)

The only change in this graph from the first graph is that phase instead of magnitude is being graphed versus Theta.



Hardcopy ---> P or p

FIG. H-11. MAGNITUDE VERSUS THETA

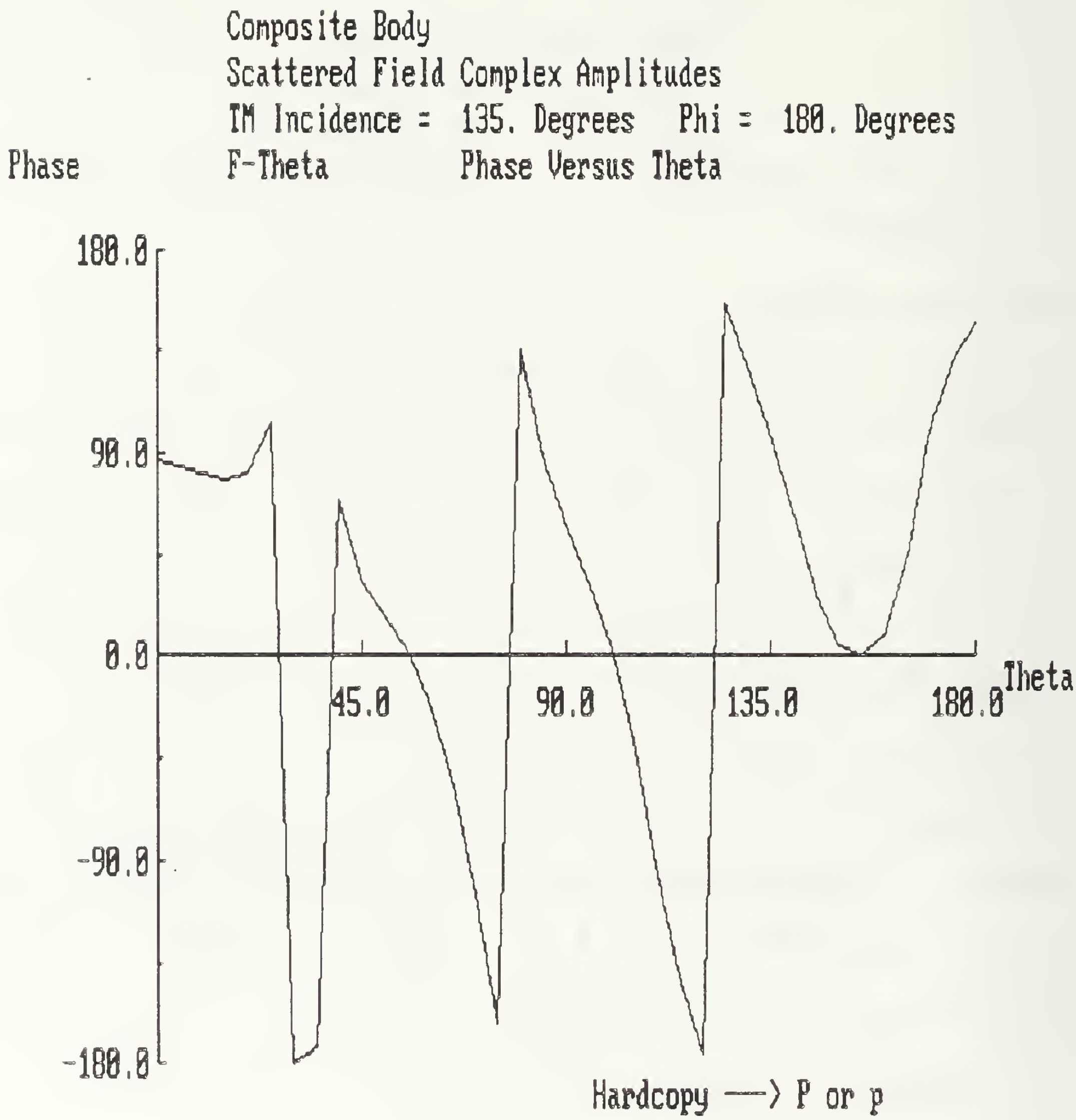


FIG. H-12. PHASE VERSUS THETA

* Magnitude Versus Theta, Polar Plot

(Figure H-13)

This graph represents the same data as the first graph, but from a different viewpoint. The first graph was drawn using rectangular coordinates. This graph utilizes polar coordinates. This graph type is the one predominately used in the industry to illustrate scattering.

* Magnitude Versus Theta, Polar Plot
  Axis Rotated Minus Ninety Degrees

(Figure H-14)

The only change in this graph from the third graph is the rotation of the axis. The limitations of GRAFMATIC made it difficult to generate the data, maintain spacing restrictions and still display all the data necessary to adequately describe the situation. Rotating the axis ninety degrees displays the data slightly differently from the third graph, giving the user a bit more useful information.

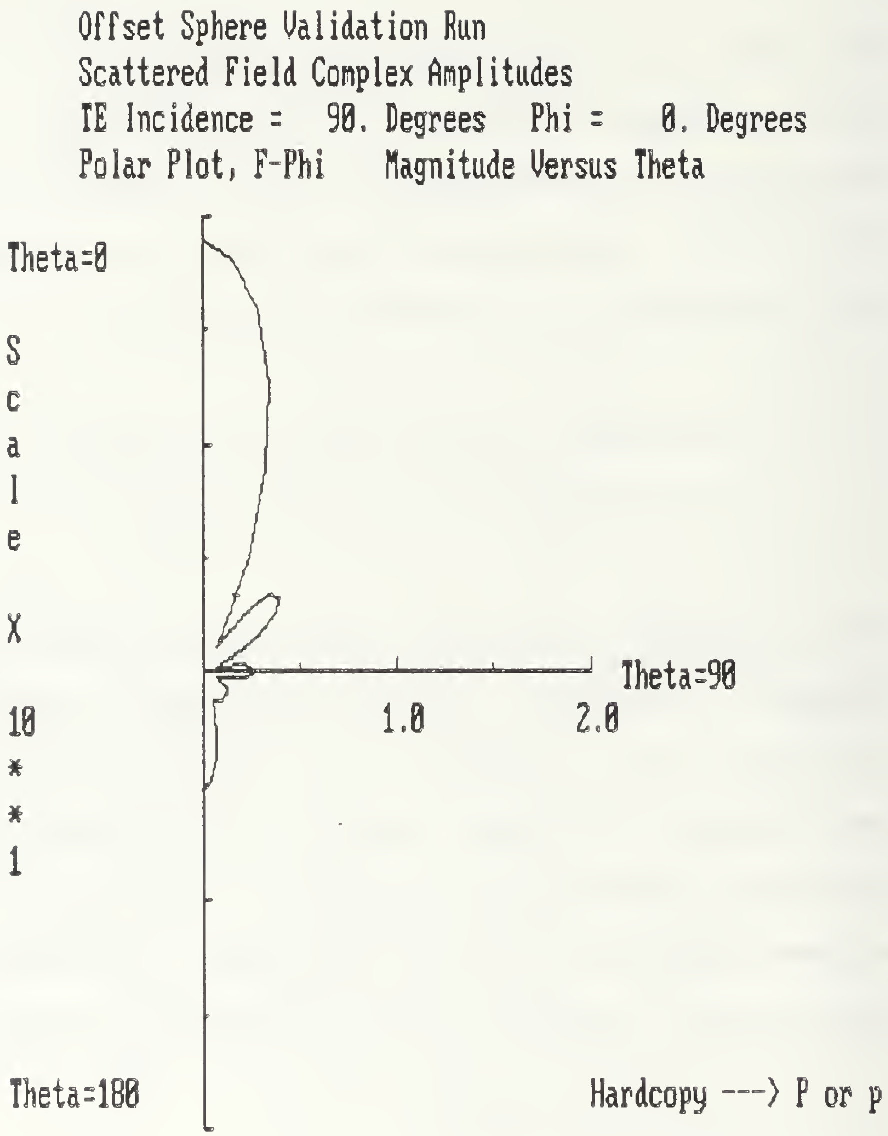


FIG. H-13. MAGNITUDE VERSUS THETA, POLAR PLOT

Composite Body
Scattered Field Complex Amplitudes
TM Incidence = 135. Degrees Phi = 180. Degrees
Polar Plot, F-Theta Magnitude Versus Theta
Axis Rotated Minus Ninety Degrees

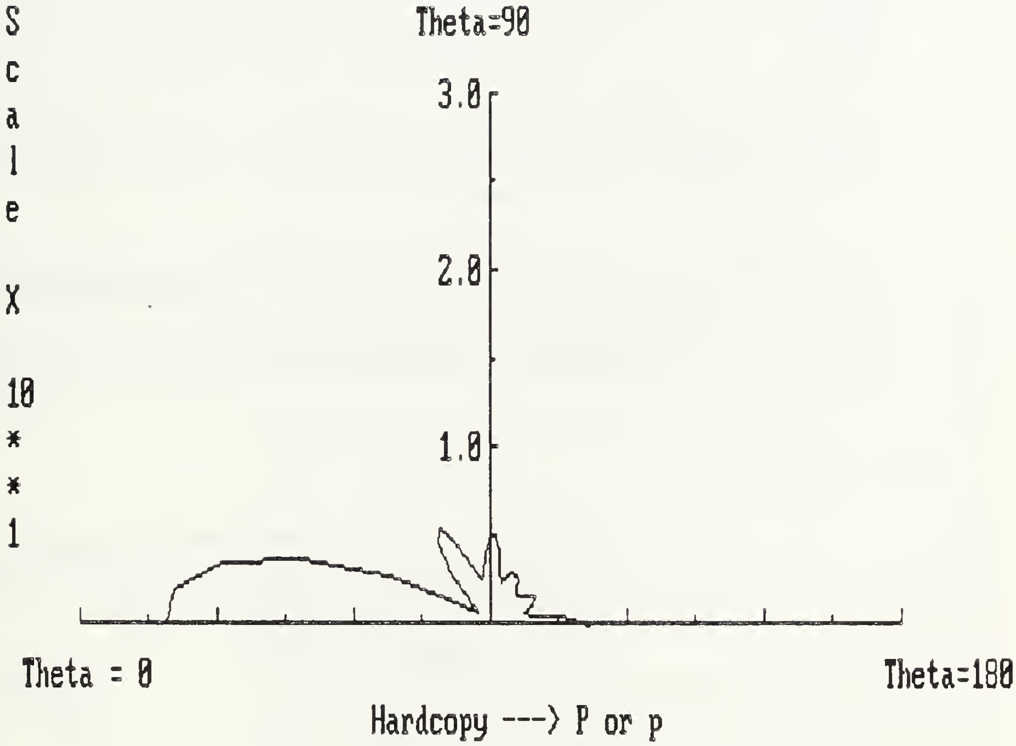


FIG. H-14. MAGNITUDE VERSUS THETA, POLAR PLOT, AXIS ROTATED MINUS NINETY DEGREES

APPENDIX I

MCADGRAF SOURCE CODE

```
C      PROGRAM MCADGRAF.FOR
C
C      *******************************************************************
C
C      THIS PROGRAM WAS WRITTEN TO BE USED IN CONJUNCTION WITH THE
C      FORTRAN PROGRAM ENTITLED EMCAD.  MCADGRAF ADDS GRAPHING FEATURES
C      TO SUPPORT THE ANALYSIS OF DATA GENERATED AND FORMATTED BY EMCAD.
C      EMCAD WAS WRITTEN BY DR. M. A. MORGAN, WITH SUBSEQUENT UPDATE BY
C      HIMSELF AND LT E. M. CONNOLLY WHILE AT THE NAVAL POSTGRADUATE
C      SCHOOL (MODS MAR 87 - MAR 88).  MCADGRAF WAS DEVELOPED AND WRITTEN
C      BY LT E. M. CONNOLLY TO HELP FULFILL THESIS REQUIREMENTS WHILE
C      UNDER THE GUIDANCE AND THE SUPPORT OF DR. MORGAN.  MCADGRAF WAS
C      WRITTEN AND COMPILED USING RYAN MCFARLAND FORTRAN, GRAFMATIC, AND
C      GRAFPLUS.  ANY QUESTIONS CONCERNING COLORS, FORMAT, CHANGES TO THE
C      PROGRAM, ETC... SHOULD BE REFERRED TO THE RESPECTIVE USERS
C      MANUALS, DR. M. A. MORGAN, OR LT E. M. CONNOLLY.
C
C      A NOTE OF CAUTION, TO OBTAIN CORRECT PRINTOUTS FROM MCADGRAF,
C      GRAFPLUS MUST BE PROPERLY CONFIGURED TO SUPPORT YOUR HARDWARE.
C      QUESTIONS CONCERNING THE PROPER HARDWARE SETUP SHOULD BE REFERRED
C      TO THE GRAFPLUS USER'S MANUAL.
C
C
C      *******************************************************************
C      *******************************************************************
C      PROGRAM STATEMENT DEFINITIONS
C      *******************************************************************
C      *******************************************************************
C
C
C
C
C        QSMODE (mode)
C
C                         Clears the screen and sets the mode of the CRT
C                         display.  May be used in either the text or the
C                         graphics mode.  The desired value of mode is
C                         chosen from the following table:
C
C                         MODE    RESOLUTION (CxR)       COLOR    T/G
C                         0       40 X 25                B/W      T
C                         1       40 X 25                C        T
C                         2       80 X 25                B/W      T
C                         3       80 X 25                C        T
C                         4       320 X 200              C        G
C                         5       320 X 200              B/W      G
C                         6       640 X 200              B/W      G
C                         7       80 X 25                B/W      T(M)
C                         13*     320 X 200              C        G
C                         14*     640 X 200              C        G
C                         15*     640 X 350              50HZ MONO
C                         16*     640 X 350              4/16 COLORS
C                         * ----> Requires EGA card
C
C
C        QPLOT ( jcol1, jcol2, jrow1, jrow2, xmin, xmax, ymin, ymax,
C                xorg, yorg, iopt, yoverx, aspect)
C
```

```
C                         Plot initialization routine: sets plot window,
C                         corresponding x and y ranges, and scaling
C                         factors.
C
C       jcol1             left hand pixel column defining plot region
C       jcol2             right hand pixel column defining plot region
C       jrow1             lower pixel row defining plot region
C       jrow2             upper pixel row defining plot region
C       xmin              value of x variable assigned to correspond to
C                         column jcol1
C       xmax              value of x variable assigned to correspond to
C                         column jcol2
C       ymin              value of y variable assigned to correspond to
C                         row jrow1
C       ymax              If iopt = 0, then ymax is the value of the y
C                         variable assigned to row = jrow2.  If iopt = 1,
C                         then on input ymax is the assumed maximum value
C                         of the y variable required by the plot.
C       xorg              vertical (y) axis intersects the horizontal (x)
C                         axis at x = xorg
C       yorg              horizontal (x) axis intersects the wertical (y)
C                         axis at y = yorg
C       iopt              sets scaling method option, see ymax
C       yoverx            number of y unit changes for a unit change in
C                         value of x, ie.. the change in the y coordinate
C                         per inch divided by the change in the x
C                         coordinate per inch
C       aspect            aspect ratio for the user's CRT, for NEC
C                         Multisync, aspect = 0.6 for mode = 16
C
C
C       QXAXIS ( xst, xfin, xmajor, minor, label, ndec)
C
C                         Draws the x axis along the horizontal direction,
C                         including tic marks and tic mark labels
C
C       xst               value of x at the start of the x axis;
C                         If a logarithmic scale is being used, then
C                         xst and xfin must be one of the following
C                         with xfin > xst :
C                         0.001, 0.01, 1.0, 10.0, 100.0, 1000.0, 10000.0
C       xfin              value of x at the end of the x axis
C       xmajor            distance between the major (large) tic marks;
C                         If xmajor is positive/negative, then the tic
C                         marks will be placed above/below the x axis
C                         respectively.  If xmajor is zero, then no tic
C                         marks are inserted
C       minor             integer flag for use of minor tic marks or
C                         logarithmic scale;  If minor = 0, then no tic
C                         marks are inserted.  If minor = +1, then minor
C                         tic marks are inserted halfway between the
C                         major tic marks, but are not labeled.  If
C                         minor = -1, then a logarithmic set of tic marks
C                         are inserted.
C       label             integer flag for auto labeling of tic marks;
C                         If label = 0, then there is no auto labeling.
C                         If label = 1, then the major tic marks are
C                         automatically supplied with a label.  If label
C                         = -1, then the origin is labeled as well.
C       ndec              number of decimal places for auto labeling;
C                         The format of the label is F10.n with the
C                         decimal point lined up with the major tic marks.
C
C
C       QYAXIS ( yst, yfin, ymajor, minor, label, ndec )
C
```

```
C                            Creates the y axis for a plot.
C
C       yst                  value of y at the start of the y axis
C       yfin                 value of y at the end of the y axis
C       ymajor               distance between major (large) tic marks on
C                            the y axis
C       minor                integer flag for use of minor tic marks or
C                            logarithmic scale, see QXAXIS
C       label                integer flag for auto labeling of tic marks,
C                            see QXAXIS
C       ndec                 number of decimal places for auto labeling,
C                            see QXAXIS, 0 <= ndec <= 3
C
C       QPREG ( ireg, ival)
C
C                            Changes the color table by modifying the
C                            Pallette Registers, EGA card only
C
C       ireg                 pallette register index (0-15)
C       ival                 value to be placed in register (0-63)
C
C
C       QPTXT ( nchar, char, kolor, icol, irow )
C
C                            Writes plot text.  The character variable char
C                            of length nchar is written at ( column, row ) =
C                            ( icol, irow ) in specified color.
C
C       nchar                length of character variable
C       char                 character variable corresponding to desired
C                            text
C       kolor                specifies color of the text
C       icol                 starting text column of text
C       irow                 text row containing text
C
C       QPTXTB ( nchar, char, kolor )
C
C                            Writes plot text at position B.  A character
C                            variable char of length nchar is written at
C                            position ("B") just beyond the positive X axis
C
C       QPTXTC ( nchar, char, kolor )
C
C                            Writes plot text at position C.  A character
C                            variable char of length nchar is written at
C                            position ("C") just above and centered on the
C                            Y axis
C
C       QCMOV ( icol, irow )
C
C                            Moves the cursor to a specified column and
C                            row
C
C       QSETUP ( ndots, lkolor, isymbl, klrsym )
C
C                            Sets up line type (solid or dashed) and symbols
C                            to be used by QTABLE when plotting data
C
C       ndots                Number of pixels (dots) specifying length of the
C                            dash. If ndots = 0, then a solid line is drawn.
C                            The input ndots = 1 is not allowed. By varying
C                            ndots > = 2, you may create a line composed of
C                            variable length dashes
C       lkolor               Specifies color of the line
C       isymbl               Specifies symbol to be inserted at data points
C                            QTABL. If isymbl = -1, then a cross is drawn at
```

```
C                         the data points. If isymbl = -2, then no symbol
C                         is drawn. Otherwise, the character with the
C                         ascii number "isymbl" is drawn
C       klrsym            Specifies color of the symbol
C
C       QTABL ( itype, npt, xarray, yarray )
C
C                         Plots a tabulated function
C
C       itype             Specifies point or line plot. When itype = 0,
C                         then the plot is a series of symbols at the
C                         tabulated points. When itype = 1, the symbols
C                         are joined by straight lines
C       npt               The number of points in the function table
C       xarray            X values in the function table
C       yarray            Y values in the function table
C       select1           A character string used to hold the input
C                         values of incidence angle and phi angle
C                         selection
C       scalexy           A scale factor used in three of the four graphs
C       xymax             The maximum scaled value out of all the scaled
C                         X values and all of the scaled Y values
C       fname1            A character string used to hold the value of
C                         the input data file name without extension
C       fname2            A character string used to hold the value of
C                         the input data file name with extension
C       xtension          A character string used to hold the value of
C                         the input data filename extension, i.e.
C                         .TET, .TEP, .TMT, or .TMP
C       grafprint         A character string used to allow the user to
C                         print out a graph from the screen
C       dummy             An input variable used to advance the screen
C       title1            The first line of the title for each graph
C       title2            The second line of the title for each graph
C       title3            The third line of the title for each graph
C       title4            The fourth line of the title for each graph
C       xarray            Data files contain numbers in THET, MAG, PHASE
C                         format.  This polar coordinate format can be
C                         transformed to rectangular coordinate format.
C                         XARRAY is the array established to hold all of
C                         the transformed X values.
C       yarray           Data files contain numbers in THET, MAG, PHASE
C                         format.  This polar coordinate format can be
C                         transformed to rectangular coordinate format.
C                         YARRAY is the array established to hold all of
C                         the transformed Y values.
C
C       xamax             The maximum value of the X array (XARRAY)
C       yamax             The maximum value of the Y array (YARRAY)
C       gamax             The maximum value of the X array (XARRAY)
C                         and the Y array (YARRAY)
C       pi                A constant approximately equal to 22/7
C       fname             A character string which equals either F-Phi or
C                         F-Theta
C       numsets           A number used for indexing purposes, equal to
C                         number of incidence angles times the number of
C                         phi angles
C       numpts            The number of theta angles per given incidence
C                         angle and phi angle
C       a,b,c             Dummy variables used for incrementation.
C                              a    Index for incidence angles
C                              b    Index for phi angles
C                              c    Index for theta angles
C       incangi           A dummy variable used to read and hold the
C                         value of the incidence angle presently being
C                         utilized
```

```
C        incangl(a)        A matrix containing the values of all the
C                          incidence angles
C        i                 A dummy integer used as a counter
C        phii              A dummy variable used to read and hold the
C                          value of the phi angle presently being utilized
C        incangld          A dummy variable containing an incidence angle
C                          value used for comparison purposes
C        thetai            A dummy variable used to read and hold the
C                          value of the theta angle presently being
C                          utilized
C        magi              The magnitude of the output for the existing
C                          incidence angle, phi angle, and theta angle
C        phasei            A dummy variable used to read and hold the
C                          value of the phase angle presently being
C                          utilized
C        title6            A dummy character string used to hold an input
C                          title to be used in an output graph
C        title7            A dummy character string used to hold an input
C                          title to be used in an output graph
C        title8            A dummy character string used to hold an input
C                          title to be used in an output graph
C        tarray(c)         A matrix containing the values of all the theta
C                          angles in degrees for a given incidence angle
C                          and phi angle
C        rmarray(c)        A matrix containing the values of all the
C                          magnitudes for a given incidence angle
C                          and phi angle
C        parray(c)         A matrix containing the values of all the
C                          phase angles for a given incidence angle
C                          and phi angle
C        calcvar           A variable used in intermediate calculatioins
C                          to determine a graphical scale factor (scalefac)
C        scalefac          A graphical scale factor
C        dummyvar          A dummy variable used as a limit in scale factor
C                          calculations for visual graphic asthetics
C
C
C        theta(a,b,c)      A matrix containing the values of all the theta
C                          angles for all the incidence angles and all the
C                          phi angles
C        mag(a,b,c)        A matrix containing the values of all the
C                          magnitudes for all the incidence angles and
C                          all the phi angles
C        phase(a,b,c)      A matrix containing the values of all the phase
C                          angles for all the incidence angles and all the
C                          phi angles
C        rad(c)            A matrix containing the values of all the theta
C                          angles in radians for a given incidence angle
C                          and phi angle
C
C     *****************************************************************
C     *****************************************************************
C     VARIABLE DECLARATIONS
C     *****************************************************************
C     *****************************************************************
C
      INTEGER A, B, C, I, IREG, IVAL, SCALEFAC
      INTEGER*2 ICOL, IOPT, IROW, ISYMBL, JCOL1, JCOL2, JROW1
      INTEGER*2 JROW2, KLRSYM, KOLOR, LABEL, LKOLOR, MINOR
      INTEGER*2 MODE, NCHAR, NDEC, SELECT1
      CHARACTER*2 SELECT, XTENSION
      CHARACTER*3 GRAFPRNT
      CHARACTER*8 DUMMY, FNAME1, TITLE7
      CHARACTER*12 FNAME2
      CHARACTER*20 CHAR
      CHARACTER*64 TITLE1, TITLE2, TITLE3, TITLE4, TITLE5
```

```
      CHARACTER*64 TITLE8, TITLE6(5,19,1)
      REAL ASPECT, CALCVAR, DUMMYVAR, GAMAX, GMAX, INCANGI
      REAL INCANGLD, PHASEI, PHII, PI, MAGI, NDOTS, NUMPTS
      REAL NUMSETS, RAD, THETAI
      REAL XAMAX, XFIN, XMAJOR, XMAX, XMIN, XORG, XST
      REAL YAMAX, YFIN, YMAJOR, YMAX, YMIN, YORG, YOVERX, YST
      REAL AARRAY(37), BARRAY(37), INCANGL(5), PHI(19)
      REAL MAG(5,19,37), PHASE(5,19,37), THETA(5,19,37)
      COMMON/ONE/ RMARRAY(37), SCALEFAC, TARRAY(37)
      COMMON/TWO/ PARRAY(37)
      COMMON/THREE/ XARRAY(37), YARRAY(37), SCALEXY, XYMAX
C
C     *****************************************************************
C     *****************************************************************
C     READ DATA FILES, LOAD ARRAYS, CALCULATE MAXIMUM VALUES
C     *****************************************************************
C     *****************************************************************
C
C     *****************************************************************
C     ESTABLISH INITIAL VALUES
      pi = 3.1416
      aspect = .65
C     *****************************************************************
C     ESTABLISH INITIAL SCREEN MODE AND COLORS
      mode = 3
      CALL qsmode(mode)
C     *****************************************************************
C     WELCOME TO MCADGRAF
      write(*,103)
      read(*,100) dummy
      CALL qsmode(mode)
C     *****************************************************************
C     NOTES CONCERNING GRAPHICAL OUTPUT PART 1
   60 write(*,101)
      read(*,100) select
      CALL qsmode(mode)
      if(select .ne. '1' .and. select .ne. '2') go to 60
      if(select .eq. '2') go to 1
C     *****************************************************************
C     NOTES CONCERNING GRAPHICAL OUTPUT PART 2
      write(*,102)
      read(*,100) dummy
      CALL qsmode(mode)
C     *****************************************************************
C     DATA FILE NAME INPUT MENU
    4 write(*,91)
      read(*,100) select
      if(select .ne. '1'.and. select .ne. '2')  go to 4
      if(select .eq. '2') go to 1
      CALL qsmode(mode)
C     *****************************************************************
C     INPUT DATA FILE NAME
   45 write(*,92)
      read(*,100) fname1
      CALL qsmode(mode)
C     *****************************************************************
C     DATA FILE TYPE SELECTION MENU
   44 write(*,90)
      read(*,100) select
      xtension = select
      CALL qsmode(mode)
      if(select .ne. '1' .and. select .ne. '2' .and. select .ne. '3'
     1.and. select .ne. '4' .and. select .ne. '5' .and. select .ne. '6'
     2.and. select .ne. '7') go to 44
      if(select .eq. '7') go to 1
      if(select .eq. '6') go to 52
```

```
      if(select .eq. '5') go to 45
C     ******************************************************************
C     OPEN DATA FILE
      if(xtension .eq. '1') then
        fname2 = fname1//'.tet'
      else if(xtension .eq. '2') then
        fname2 = fname1//'.tep'
      else if(xtension .eq. '3') then
        fname2 = fname1//'.tmt'
      else
        fname2 = fname1//'.tmp'
      endif
      CALL qsmode(mode)
      open(4, file=fname2)
C     ******************************************************************
C     READ DATA FILE
      read(4,100) title8
      read(4,100) title7
      read(4,86) numsets
      read(4,86) numpts
      a = 1
      incangld = -1000.
      b = 1
      do 16 i = 1,int(numsets)
      c= 1
      read(4,86) incangi
      if(incangi .ne. incangld) b = 1
      if(incangi .ne. incangld) a = a + 1
      if(i .eq. 1) a = 1
      incangl(a) = incangi
      incangld = incangi
      read(4,86) phii
      phi(b) = phii
      read(4,100) title6(a,b,c)
      do 17 c = 1,int(numpts)
      read(4,88) thetai, magi, phasei
      theta(a,b,c) = thetai
      mag(a,b,c) = magi
      phase(a,b,c) = phasei
   17 continue
      b = b + 1
   16 continue
      close(4)
C     ******************************************************************
C     SELECT DATA FILE SUBSET
C     (I.E.   SELECT INCIDENCE ANGLE AND PHI ANGLE)
    5 write(*,93) a
      do 25 i=1,a
        write(*,94) i,incangl(i)
   25 continue
      write(*,95) a+1, a+2
      write(*,96)
      read(*,*) select1
      CALL qsmode(mode)
      if(select1 .ne. 1 .and. select1 .ne. 2 .and. select1 .ne. 3
     1.and. select1 .ne. 4 .and. select1 .ne. 5 .and. select1 .ne. 6
     2 .and. select1 .ne. 7) go to 5
      if(select1 .eq. a+1) go to 44
      if(select1 .eq. a+2) go to 1
      a = select1
    6 write(*,97) b-1
      do 26 i=1,b-1
        write(*,98) i,phi(i)
   26 continue
      write(*,99) b, b+1
      write(*,96)
```

```
      read(*,*) selectl
      CALL qsmode(mode)
      CALL qpreg(0,32)
      if(selectl .eq. b) go to 44
      if(selectl .eq. b+1) go to 1
      b = selectl
C     ****************************************************************
C     LOAD ARRAYS
      c = 1
      xamax = 0.0
      yamax = 0.0
      gamax = 0.0
      title2 = title6(a,b,c)
      do 18 c = 1,int(numpts)
      tarray(c) = theta(a,b,c)
      rmarray(c) = mag(a,b,c)
      parray(c) = phase(a,b,c)
      rad = tarray(c) * pi / 180
      xarray(c) = rmarray(c) * sin(rad)
      yarray(c) = rmarray(c) * cos(rad)
C     ****************************************************************
C     CALCULATE MAXIMUM VALUES
      if (abs(xarray(c)) .gt. xamax) xamax = abs(xarray(c))
      if (abs(yarray(c)) .gt. yamax) yamax = abs(yarray(c))
   18 continue
   75 continue
C     ****************************************************************
C     ****************************************************************
C     CALCULATE SCALE FACTORS
      gamax = abs(xamax)
      if (yamax .gt. xamax) gamax = abs(yamax)
      if (abs(gamax - 0) .lt. .000001) gamax = .000001
      calcvar = alog10(gamax)
      scalexy = int(calcvar)
      if (gamax .lt. 1.0) scalexy = int(calcvar) - 1
      scalefac = int(scalexy)
      dummyvar = gamax/(10**scalexy)
      xymax = int(dummyvar + 1)
C     ****************************************************************
C     CLEAR SCREEN
      mode = 16
      CALL qsmode(mode)
C     ****************************************************************
      CALL rectmvt(title2,title7,title8,numpts,aspect)
C     ****************************************************************
C     CLEAR SCREEN
      mode = 16
      CALL qsmode(mode)
C     ****************************************************************
      CALL rectpvt(title2,title7,title8,numpts,aspect)
C     ****************************************************************
C     CLEAR SCREEN
      mode = 16
      CALL qsmode(mode)
C     ****************************************************************
      CALL polarmvt(title2,title7,title8,numpts,aspect)
C     ****************************************************************
C     CLEAR SCREEN
      mode = 16
      CALL qsmode(mode)
C     ****************************************************************
      CALL polarrot(title2,title7,title8,numpts,aspect)
C     ****************************************************************
C     CLEAR SCREEN
      mode = 16
      CALL qsmode(mode)
```

```
C     ***************************************************************
      goto 44
C     ***************************************************************
C     ASPECT RATIO
   52 write(*,104)
      read(*,*) aspect
      mode = 3
      CALL qsmode(mode)
      goto 44
C     ***************************************************************
    1 continue
C     CLEAR SCREEN
      mode = 3
      CALL qsmode(mode)
C     ***************************************************************
   80 format(6X,F5.0,3X,E11.3,6X,F7.2)
   86 format(F5.0)
   88 format(F5.0,E11.3,F7.2)
   90 format(/5X,'EMCAD OUTPUT GRAPHICS PRESENTATION',/5X,
     1'DATA FILE TYPE SELECTION MENU',//5X,
     2'Please select the type of data file containing the information of
     3',/5X,
     4'interest.  EMCAD automatically created these four data file type
     5s.',//8X,
     6'1.   TE INCIDENCE,    F-THETA',//8X,
     7'2.   TE INCIDENCE,    F-PHI  ',//8X,
     8'3.   TM INCIDENCE,    F-THETA',//8X,
     9'4.   TM INCIDENCE,    F-PHI  ',//8X,
     1'5.   CHANGE DATA FILE',//8X,
     2'6.   CHANGE ASPECT RATIO',//8X,
     3'7.   EXIT THE PROGRAM',//5X,
     4'Indicate your selection by the choice of 1, 2, 3, 4, 5, 6 or 7',
     5//5X)
   91 format(//5X,'EMCAD OUTPUT GRAPHICS PRESENTATION',//5X,
     1'DATA FILE NAME INPUT MENU',//5X,
     2'If utilizing a file or series of files created by EMCAD,',/5X,
     3'please select option 1.  If utilizing any file other than',/5X,
     4'one created by EMCAD, please ensure that the file is in',/5X,
     5'proper format, and that the filename extension is one of',/5X,
     6'the four extensions utilized by this program.  These',/5X,
     7'extensions are:  .tet   .tep   .tmt   and   .tmp',/5X,
     8'If you are unsure as to whether your file is in the proper',/5X,
     9'format with the proper extension, please exit the program',/5X,
     1'and verify before continuing.',//8X,
     2'1.   INPUT THE FILE OF USER CHOICE',//8X,
     3'2.   EXIT THE PROGRAM',//5X,
     4'Indicate your selection by the choice of 1 or 2.'//5X)
   92 format(//5X,'EMCAD OUTPUT GRAPHICS PRESENTATION',//5X,
     1'DATA FILE NAME INPUT ',//5X,
     2'Please input the name of the file containing data that',//5X,
     3'you wish to work with.  Please input only the file name',//5X,
     4'i.e.  do not include the file extension.  The file',//5X,
     5'extension will be selected in the DATA FILE TYPE',//5X,
     6'SELECTION MENU.',//5X)
   93 format(//5X,'EMCAD OUTPUT GRAPHICS PRESENTATION',//5X,
     1'SELECTION OF INCIDENCE ANGLE',/5X,
     2'The data file you have selected contains data for ',I2,/5X,
     3'incidence angles.  Please select your choice from the',/5X,
     4'following menu.  Note, you may only work with one',/5X,
     5'incidence angle at a time.',/)
   94 format(5X,I1,5X,'INCIDENCE ANGLE =  ',F6.2)
   95 format(5X,I1,5X,
     1'RETURN TO THE DATA FILE TYPE SELECTION MENU',/5X,I1,5X,
     2'EXIT THE PROGRAM')
   96 format(/5X,'INDICATE YOUR SELECTION BY THE CHOICE OF A NUMBER  I.E
     1. 1  2  3 ...)',//5X)
```

```
 97 format(/5X,'EMCAD OUTPUT GRAPHICS PRESENTATION',//5X,
   1'SELECTION OF PHI ANGLE',/5X,
   2'The data file you have selected contains data for ',I2,/5X,
   3'phi angles.  Please select your choice from the following',/5X,
   4'menu.  Note, you may only work with one phi angle at a time.',/)
 98 format(5X,I2,5X,'PHI ANGLE =  ',F6.2)
 99 format(5X,I2,5X,
   1'RETURN TO THE DATA FILE TYPE SELECTION MENU',/5X,I2,5X,
   2'EXIT THE PROGRAM')
100 format(A)
101 format(/5X,'EMCAD OUTPUT GRAPHICS PRESENTATION',//5X,
   1'NOTES CONCERNING GRAPHICAL HARDCOPY OUTPUT,  PART 1',//5X,
   2'This program allows the user to obtain printer hardcopies',/5X,
   3'of the graphs being viewed.  Hardcopy graphic outputs are',/5X,
   4'obtained through the use of the program entitled GRAFPLUS.',/5X,
   5'Proper utilization of MCADGRAF requires prior setup of',/5X,
   6'GRAFPLUS.  Directions for proper setup are contained',/5X,
   7'in the GRAFPLUS users manual.',//5X,
   8'1.      CONTINUE WITH THE NEXT SCREEN',//5X,
   9'2.      EXIT THE PROGRAM',//5X,
   1'Indicate your selection by the choice of 1 or 2.'//5X)
102 format(/5X,'EMCAD OUTPUT GRAPHICS PRESENTATION',//5X,
   1'NOTES CONCERNING GRAPHICAL HARDCOPY OUTPUT,  PART 2',//5X,
   2'The user makes the decision as to whether a printer ',/5X,
   3'graphical output is desired after viewing the graph.  Once',/5X,
   4'a graph is drawn on the screen, the user may request a ',/5X,
   5'printer hardcopy by typing either a "P" or "p" and pressing',/5X,
   6'the "ENTER" key.  Selection of any other key will simply',/5X,
   7'advance to the next graph.  Spacing on 8 1/2 by 11 paper ',/5X,
   8'allows up to 2 graphs per page.',///5X,
   9'PLEASE PRESS ENTER TO CONTINUE.',5X)
103 format(/5X,'********** WELCOME TO MCADGRAF **********',//5X,
   1'THIS PROGRAM WAS WRITTEN TO BE USED IN CONJUNCTION WITH ',/5X,
   2'THE FORTRAN PROGRAM ENTITLED EMCAD.  MCADGRAF ADDS ',/5X,
   3'GRAPHING FEATURES TO SUPPORT THE ANALYSIS OF DATA ',/5X,
   4'GENERATED AND FORMATTED BY EMCAD.  EMCAD WAS WRITTEN BY',/5X,
   5'DR. M. A. MORGAN, WITH MODIFICATIONS BY HIMSELF AND ',/5X,
   6'LT E. M. CONNOLLY.  MCADGRAF WAS DEVELOPED AND WRITTEN BY',/5X,
   7'LT E. M. CONNOLLY TO HELP FULFILL THESIS REQUIREMENTS WHILE',/5X,
   8'UNDER THE GUIDANCE AND THE SUPPORT OF DR. MORGAN.  MCADGRAF',/5X,
   9'WAS WRITTEN AND COMPILED USING RYAN MCFARLAND FORTRAN,',/5X,
   1'GRAFMATIC, AND GRAFPLUS.  ANY QUESTIONS CONCERNING COLORS,',/5X,
   2'FORMAT, CHANGES TO THE PROGRAM, ETC... SHOULD BE REFERRED ',/5X,
   3'TO THE RESPECTIVE USERS MANUALS, DR. M. A. MORGAN, OR',/5X,
   4'LT E. M. CONNOLLY.',//5X,
   5'PLEASE PRESS ENTER TO CONTINUE.',5X)
104 format(/5X,'********** ASPECT RATIO **********',//5X,
   1'IN ORDER TO AVOID DISTORTION ON THE SCREEN, IT IS NECESSARY',/5X,
   2'TO DEFINE THE NUMBER OF ROWS PER INCH AND THE NUMBER OF',/5X,
   3'COLUMNS PER INCH.  THESE DEFINITIONS CHANGE AS THE',/5X,
   4'RESOLUTION OF THE SCREEN, AS WELL AS THE TYPE OF SCREEN',/5X,
   5'CHANGE.  THIS PROGRAM WAS WRITTEN IN MODE 16 (EGA) USING',/5X,
   6'A GB-1 VIDEO BOARD AND A NEC MULTISYNC MONITOR.  THE',/5X,
   7'OPTIMUM ASPECT RATIO FOR THIS CONFIGURATION WAS .65',///5X,
   8'PLEASE INPUT ASPECT RATIO.',//5X)
105 format(F5.3)
    end

    SUBROUTINE RECTMVT(title2,title7,title8,numpts,aspect)
C   ****************************************************************
C   ****************************************************************
C   MAGNITUDE VERSUS THETA
C   RECTANGULAR COORDINATE GRAPH
C   ****************************************************************
C   ****************************************************************
C
```

```
C     ****************************************************************
C     VARIABLE DECLARATIONS
C     ****************************************************************
      INTEGER A, B, C, I, IREG, IVAL, SCALEFAC
      INTEGER*2 ICOL, IOPT, IROW, ISYMBL, JCOL1, JCOL2, JROW1
      INTEGER*2 JROW2, KLRSYM, KOLOR, LABEL, LKOLOR, MINOR
      INTEGER*2 MODE, NCHAR, NDEC, SELECT1
      CHARACTER*2 SELECT, XTENSION
      CHARACTER*3 GRAFPRNT
      CHARACTER*8 DUMMY, FNAME1, TITLE7
      CHARACTER*12 FNAME2
      CHARACTER*20 CHAR
      CHARACTER*64 TITLE1, TITLE2, TITLE3, TITLE4, TITLE5
      CHARACTER*64 TITLE8, TITLE6(5,19,1)
      REAL ASPECT, CALCVAR, DUMMYVAR, GAMAX, GMAX, INCANGI
      REAL INCANGLD, PHASEI, PHII, PI, MAGI, NDOTS, NUMPTS
      REAL NUMSETS, RAD, THETAI
      REAL XAMAX, XFIN, XMAJOR, XMAX, XMIN, XORG, XST
      REAL YAMAX, YFIN, YMAJOR, YMAX, YMIN, YORG, YOVERX, YST
      REAL AARRAY(37), BARRAY(37), INCANGL(5), PHI(19)
      REAL MAG(5,19,37), PHASE(5,19,37), THETA(5,19,37)
      COMMON/ONE/ RMARRAY(37), SCALEFAC, TARRAY(37)
      COMMON/TWO/ PARRAY(37)
      COMMON/THREE/ XARRAY(37), YARRAY(37), SCALEXY, XYMAX
C     ****************************************************************
C     SET DISPLAY MODE
      mode = 16
      CALL qsmode(mode)
C     ****************************************************************
C     SET AXIS COLOR
      CALL qpreg(7,6)
C     ****************************************************************
C     DEFINE GRAPH REGION
      jcol1 = 080
      jcol2 = 520
      jrow1 = 40
      jrow2 = 320
      xmin = -0.5
      xmax = 180
      ymin = -0.5
      ymax = xymax
      xorg = 0.0
      yorg = 0.0
      iopt = 1
      yoverx = .7*(ymax-ymin)/(xmax-xmin)
      CALL qplot(jcol1,jcol2,jrow1,jrow2,xmin,xmax,ymin,ymax,xorg,
     1yorg,iopt,yoverx,aspect)
C     ****************************************************************
C     SET UP LINE TYPE
      ndots = 0
      lkolor = 3
      isymbl = -2
      klrsym = 4
      CALL qsetup(ndots,lkolor,isymbl,klrsym)
C     ****************************************************************
C     DRAW X AXIS
      xst = 0.0
      xfin = 180
      xmajor = 45
      minor = 0
      label = -1
      ndec = 1
      CALL qxaxis(xst,xfin,xmajor,minor,label,ndec)
C     ****************************************************************
C     DRAW Y AXIS
      yst = 0.0
```

```
      yfin = xymax
      ymajor = 2
      if(xymax .eq. 1 .or. xymax .eq. 2 .or. xymax .eq. 3 .or. xymax
     1.eq. 4 .or. xymax .eq. 5) ymajor = 1
      minor = 1
      label = -1
      CALL qyaxis(yst,yfin,ymajor,minor,label,ndec)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 20
      char = 'Theta'
      kolor = 14
      CALL qptxtb(nchar, char, kolor)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 20
      char = 'Magnitude'
      kolor = 14
      CALL qptxtc(nchar, char, kolor)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      kolor = 10
      icol = 15
      irow = 24
      CALL qptxt(nchar,title8,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      title1 = 'Scattered Field Complex Amplitudes'
      kolor = 10
      icol = 15
      irow = 23
      CALL qptxt(nchar,title1,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      kolor = 10
      icol = 15
      irow = 22
      CALL qptxt(nchar,title2,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 7
      kolor = 10
      icol = 15
      irow = 21
      CALL qptxt(nchar,title7,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 40
      title3 = ' Magnitude Versus Theta'
      kolor = 10
      icol = 30
      irow = 21
      CALL qptxt(nchar,title3,kolor,icol,irow)
C     ******************************************************************
C     SCALE "Y" VALUES
      do 30 i = 1,numpts
      aarray(i) = rmarray(i)/(10**scalexy)
   30 continue
C     ******************************************************************
C     GRAPH DATA POINTS
      itype = 1
      npt = numpts
      CALL qtabl(itype,npt,tarray,aarray)
```

```
C     ****************************************************************
C     WRITE SCALE FACTOR ON SCREEN
      nchar = 1
      title4 = 'S'
      kolor = 14
      icol = 1
      irow = 18
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'c'
      irow = 17
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'a'
      irow = 16
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'l'
      irow = 15
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'e'
      irow = 14
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'X'
      irow = 12
      CALL qptxt(nchar,title4,kolor,icol,irow)
      nchar = 2
      title4 = '10'
      irow = 10
      CALL qptxt(nchar,title4,kolor,icol,irow)
      nchar = 1
      title4 = '*'
      irow = 9
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = '*'
      irow = 8
      CALL qptxt(nchar,title4,kolor,icol,irow)
      CALL qcmov(2,8)
      write(*,11) scalefac
C     ****************************************************************
C     HARD COPY OF GRAPH ?
      nchar = 20
      title3 = 'Hardcopy ---> P or p'
      kolor = 14
      icol = 1
      irow = 1
      CALL qptxt(nchar,title3,kolor,icol,irow)
      CALL qcmov(25,1)
      read(*,100) grafprnt
      nchar = 40
      title3 = '                                        '
      kolor = 10
      icol = 1
      irow = 1
      CALL qptxt(nchar,title3,kolor,icol,irow)
      if(grafprnt .eq. 'P' .or. grafprnt .eq. 'p') CALL prtsc
C     ****************************************************************
   11 format(I3)
  100 format(A)
      return
      end

      SUBROUTINE RECTPVT(title2,title7,title8,numpts,aspect)
C     ****************************************************************
C     ****************************************************************
C     PHASE ANGLE VERSUS THETA
C     RECTANGULAR COORDINATE GRAPH
C     ****************************************************************
C     ****************************************************************
```

```
C
C     *****************************************************************
C     VARIABLE DECLARATIONS
C     *****************************************************************
      INTEGER A, B, C, I, IREG, IVAL, SCALEFAC
      INTEGER*2 ICOL, IOPT, IROW, ISYMBL, JCOL1, JCOL2, JROW1
      INTEGER*2 JROW2, KLRSYM, KOLOR, LABEL, LKOLOR, MINOR
      INTEGER*2 MODE, NCHAR, NDEC, SELECT1
      CHARACTER*2 SELECT, XTENSION
      CHARACTER*3 GRAFPRNT
      CHARACTER*8 DUMMY, FNAME1, TITLE7
      CHARACTER*12 FNAME2
      CHARACTER*20 CHAR
      CHARACTER*64 TITLE1, TITLE2, TITLE3, TITLE4, TITLE5
      CHARACTER*64 TITLE8, TITLE6(5,19,1)
      REAL ASPECT, CALCVAR, DUMMYVAR, GAMAX, GMAX, INCANGI
      REAL INCANGLD, PHASEI, PHII, PI, MAGI, NDOTS, NUMPTS
      REAL NUMSETS, RAD, THETAI
      REAL XAMAX, XFIN, XMAJOR, XMAX, XMIN, XORG, XST
      REAL YAMAX, YFIN, YMAJOR, YMAX, YMIN, YORG, YOVERX, YST
      REAL AARRAY(37), BARRAY(37), INCANGL(5), PHI(19)
      REAL MAG(5,19,37), PHASE(5,19,37), THETA(5,19,37)
      COMMON/ONE/ RMARRAY(37), SCALEFAC, TARRAY(37)
      COMMON/TWO/ PARRAY(37)
      COMMON/THREE/ XARRAY(37), YARRAY(37), SCALEXY, XYMAX
C     *****************************************************************
C     SET DISPLAY MODE
      mode = 16
      CALL qsmode(mode)
C     *****************************************************************
C     SET AXIS COLOR
      CALL qpreg(7,6)
C     *****************************************************************
C     DEFINE GRAPH REGION
      jcol1 = 080
      jcol2 = 530
      jrow1 = 30
      jrow2 = 320
      xmin = -0.5
      xmax = 180
      ymin = -180
      ymax = 180
      xorg = 0.0
      yorg = 0.0
      iopt = 1
      yoverx = 1.3
      CALL qplot(jcol1,jcol2,jrow1,jrow2,xmin,xmax,ymin,ymax,xorg,
     1yorg,iopt,yoverx,aspect)
C     *****************************************************************
C     SET UP LINE TYPE
      ndots = 0
      lkolor = 3
      isymbl = -2
      klrsym = 4
      CALL qsetup(ndots,lkolor,isymbl,klrsym)
C     *****************************************************************
C     DRAW X AXIS
      xst = 0.0
      xfin = 180
      xmajor = 45
      minor = 0
      label = 1
      ndec = 1
      CALL qxaxis(xst,xfin,xmajor,minor,label,ndec)
C     *****************************************************************
C     DRAW Y AXIS
```

```
      yst = -180
      yfin = 180
      ymajor = 90
      minor = 1
      label = -1
      CALL qyaxis(yst,yfin,ymajor,minor,label,ndec)
C     ****************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 20
      char = 'Theta'
      kolor = 14
      CALL qptxtb(nchar, char, kolor)
C     ****************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 20
      char = 'Phase'
      kolor = 14
      CALL qptxtc(nchar, char, kolor)
C     ****************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      kolor = 10
      icol = 15
      irow = 24
      CALL qptxt(nchar,title8,kolor,icol,irow)
C     ****************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      title1 = 'Scattered Field Complex Amplitudes'
      kolor = 10
      icol = 15
      irow = 23
      CALL qptxt(nchar,title1,kolor,icol,irow)
C     ****************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      kolor = 10
      icol = 15
      irow = 22
      CALL qptxt(nchar,title2,kolor,icol,irow)
C     ****************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 7
      kolor = 10
      icol = 15
      irow = 21
      CALL qptxt(nchar,title7,kolor,icol,irow)
C     ****************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 40
      title3 = ' Phase Versus Theta'
      kolor = 10
      icol = 30
      irow = 21
      CALL qptxt(nchar,title3,kolor,icol,irow)
C     ****************************************************************
C     GRAPH DATA POINTS
      itype = 1
      npt = numpts
      CALL qtabl(itype,npt,tarray,parray)
C     ****************************************************************
C     HARD COPY OF GRAPH ?
      nchar = 20
      title3 = 'Hardcopy ---> P or p'
      kolor = 14
      icol = 40
```

```
      irow = 1
      CALL qptxt(nchar,title3,kolor,icol,irow)
      CALL qcmov(65,1)
      read(*,100) grafprnt
      nchar = 40
      title3 = '                                        '
      kolor = 10
      icol = 40
      irow = 1
      CALL qptxt(nchar,title3,kolor,icol,irow)
      if(grafprnt .eq. 'P' .or. grafprnt .eq. 'p') CALL prtsc
C     ******************************************************************
  100 format(A)
      return
      end

      SUBROUTINE POLARMVT(title2,title7,title8,numpts,aspect)
C     ******************************************************************
C     ******************************************************************
C     MAGNITUDE VERSUS THETA
C     POLAR COORDINATE GRAPH
C     ******************************************************************
C     ******************************************************************
C
C     ******************************************************************
C     VARIABLE DECLARATIONS
C     ******************************************************************
      INTEGER A, B, C, I, IREG, IVAL, SCALEFAC
      INTEGER*2 ICOL, IOPT, IROW, ISYMBL, JCOL1, JCOL2, JROW1
      INTEGER*2 JROW2, KLRSYM, KOLOR, LABEL, LKOLOR, MINOR
      INTEGER*2 MODE, NCHAR, NDEC, SELECT1
      CHARACTER*2 SELECT, XTENSION
      CHARACTER*3 GRAFPRNT
      CHARACTER*8 DUMMY, FNAME1, TITLE7
      CHARACTER*12 FNAME2
      CHARACTER*20 CHAR
      CHARACTER*64 TITLE1, TITLE2, TITLE3, TITLE4, TITLE5
      CHARACTER*64 TITLE8, TITLE6(5,19,1)
      REAL ASPECT, CALCVAR, DUMMYVAR, GAMAX, GMAX, INCANGI
      REAL INCANGLD, PHASEI, PHII, PI, MAGI, NDOTS, NUMPTS
      REAL NUMSETS, RAD, THETAI
      REAL XAMAX, XFIN, XMAJOR, XMAX, XMIN, XORG, XST
      REAL YAMAX, YFIN, YMAJOR, YMAX, YMIN, YORG, YOVERX, YST
      REAL AARRAY(37), BARRAY(37), INCANGL(5), PHI(19)
      REAL MAG(5,19,37), PHASE(5,19,37), THETA(5,19,37)
      COMMON/ONE/ RMARRAY(37), SCALEFAC, TARRAY(37)
      COMMON/TWO/ PARRAY(37)
      COMMON/THREE/ XARRAY(37), YARRAY(37), SCALEXY, XYMAX
C     ******************************************************************
C     SET DISPLAY MODE
      mode = 16
      CALL qsmode(mode)
C     ******************************************************************
C     SET AXIS COLOR
      CALL qpreg(7,6)
C     ******************************************************************
C     SCALE "X" AND "Y" VALUES
      do 10 i = 1,numpts
      aarray(i) = xarray(i)/(10**scalexy)
      barray(i) = yarray(i)/(10**scalexy)
   10 continue
C     ******************************************************************
C     DEFINE GRAPH REGION
      jcol1 = 080
      jcol2 = 400
      jrow1 = 10
```

```
      jrow2 = 280
      xmin = -xymax/2
      xmax = xymax
      ymin = -xymax
      ymax = xymax
      xorg = 0.0
      yorg = 0.0
      iopt = 1
      yoverx = 0.55
      CALL qplot(jcol1,jcol2,jrow1,jrow2,xmin,xmax,ymin,ymax,xorg,
     1yorg,iopt,yoverx,aspect)
C     ******************************************************************
C     SET UP LINE TYPE
      ndots = 0
      lkolor = 3
      isymbl = -2
      klrsym = 4
      CALL qsetup(ndots,lkolor,isymbl,klrsym)
C     ******************************************************************
C     DRAW X AXIS
      xst = 0.0
      xfin = xymax
      xmajor = 2
      if(xymax .le. 5) xmajor = 1
      minor = 1
      label = 1
      ndec = 1
      CALL qxaxis(xst,xfin,xmajor,minor,label,ndec)
C     ******************************************************************
C     DRAW Y AXIS
      yst = -xymax
      yfin = xymax
      ymajor = xmajor
      label = 0
      CALL qyaxis(yst,yfin,ymajor,minor,label,ndec)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 10
      char = 'Theta=90'
      kolor = 14
      CALL qptxtb( nchar, char, kolor)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 10
      char = 'Theta=0'
      kolor = 14
      icol = 10
      irow = 19
      CALL qptxt( nchar, char, kolor, icol, irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 10
      char = 'Theta=180'
      kolor = 14
      icol = 10
      irow = 1
      CALL qptxt( nchar, char, kolor, icol, irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      kolor = 10
      icol = 15
      irow = 24
      CALL qptxt( nchar, title8, kolor, icol, irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
```

```
      nchar = 64
      title1 = 'Scattered Field Complex Amplitudes'
      kolor = 10
      icol = 15
      irow = 23
      CALL qptxt( nchar, title1, kolor, icol, irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      kolor = 10
      icol = 15
      irow = 22
      CALL qptxt(nchar,title2,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 11
      title3 = 'Polar Plot,'
      kolor = 10
      icol = 15
      irow = 21
      CALL qptxt(nchar,title3,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 7
      kolor = 10
      icol = 27
      irow = 21
      CALL qptxt(nchar,title7,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 40
      title3 = ' Magnitude Versus Theta'
      kolor = 10
      icol = 35
      irow = 21
      CALL qptxt(nchar,title3,kolor,icol,irow)
C     ******************************************************************
C     WRITE SCALE FACTOR ON SCREEN
      nchar = 1
      title4 = 'S'
      kolor = 14
      icol = 10
      irow = 17
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'c'
      irow = 16
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'a'
      irow = 15
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'l'
      irow = 14
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'e'
      irow = 13
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'X'
      irow = 11
      CALL qptxt(nchar,title4,kolor,icol,irow)
      nchar = 2
      title4 = '10'
      irow = 9
      CALL qptxt(nchar,title4,kolor,icol,irow)
      nchar = 1
      title4 = '*'
      irow = 8
```

```
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = '*'
      irow = 7
      CALL qptxt(nchar,title4,kolor,icol,irow)
      CALL qcmov(5,7)
      write(*,12) scalefac
C     See discussion of QCMOV for problem requiring this fix
C     *****************************************************************
C     GRAPH DATA POINTS
      itype = 1
      npt = numpts
      CALL qtabl(itype,npt,aarray,barray)
C     *****************************************************************
C     HARD COPY OF GRAPH ?
      nchar = 20
      title3 = 'Hardcopy ---> P or p'
      kolor = 14
      icol = 50
      irow = 1
      CALL qptxt(nchar,title3,kolor,icol,irow)
      CALL qcmov(75,1)
      read(*,100) grafprnt
      nchar = 26
      title3 = '                          '
      kolor = 10
      icol = 50
      irow = 1
      CALL qptxt(nchar,title3,kolor,icol,irow)
      if(grafprnt .eq. 'P' .or. grafprnt .eq. 'p') CALL prtsc
C     *****************************************************************
   12 format(9x,I3)
  100 format(A)
      return
      end

      SUBROUTINE POLARROT(title2,title7,title8,numpts,aspect)
C     *****************************************************************
C     *****************************************************************
C     MAGNITUDE VERSUS THETA
C     POLAR COORDINATE GRAPH
C     AXIS ROTATED MINUS NINETY DEGREES
C     *****************************************************************
C     *****************************************************************
C
C     *****************************************************************
C     VARIABLE DECLARATIONS
C     *****************************************************************
      INTEGER A, B, C, I, IREG, IVAL, SCALEFAC
      INTEGER*2 ICOL, IOPT, IROW, ISYMBL, JCOL1, JCOL2, JROW1
      INTEGER*2 JROW2, KLRSYM, KOLOR, LABEL, LKOLOR, MINOR
      INTEGER*2 MODE, NCHAR, NDEC, SELECT1
      CHARACTER*2 SELECT, XTENSION
      CHARACTER*3 GRAFPRNT
      CHARACTER*8 DUMMY, FNAME1, TITLE7
      CHARACTER*12 FNAME2
      CHARACTER*20 CHAR
      CHARACTER*64 TITLE1, TITLE2, TITLE3, TITLE4, TITLE5
      CHARACTER*64 TITLE8, TITLE6(5,19,1)
      REAL ASPECT, CALCVAR, DUMMYVAR, GAMAX, GMAX, INCANGI
      REAL INCANGLD, PHASEI, PHII, PI, MAGI, NDOTS, NUMPTS
      REAL NUMSETS, RAD, THETAI
      REAL XAMAX, XFIN, XMAJOR, XMAX, XMIN, XORG, XST
      REAL YAMAX, YFIN, YMAJOR, YMAX, YMIN, YORG, YOVERX, YST
      REAL AARRAY(37), BARRAY(37), INCANGL(5), PHI(19)
      REAL MAG(5,19,37), PHASE(5,19,37), THETA(5,19,37)
      COMMON/ONE/ RMARRAY(37), SCALEFAC, TARRAY(37)
```

```
      COMMON/TWO/ PARRAY(37)
      COMMON/THREE/ XARRAY(37), YARRAY(37), SCALEXY, XYMAX
C     ***************************************************************
C     SET DISPLAY MODE
      mode = 16
      CALL qsmode(mode)
C     ***************************************************************
C     SET AXIS COLOR
      CALL qpreg(7,6)
C     ***************************************************************
C     DEFINE GRAPH REGION
      jcol1 = 080
      jcol2 = 530
      jrow1 = 10
      jrow2 = 320
      xmin = -xymax
      xmax = xymax
      ymin = -xymax/4
      ymax = xymax
      xorg = 0.0
      yorg = 0.0
      iopt = 1
      yoverx = 0.5
      CALL qplot(jcol1,jcol2,jrow1,jrow2,xmin,xmax,ymin,ymax,xorg,
     1yorg,iopt,yoverx,aspect)
C     ***************************************************************
C     SET UP LINE TYPE
      ndots = 0
      lkolor = 3
      isymbl = -2
      klrsym = 4
      CALL qsetup(ndots,lkolor,isymbl,klrsym)
C     ***************************************************************
C     DRAW X AXIS
      xst = -xymax
      xfin = xymax
      xmajor = 2
      if(xymax .le. 5) xmajor = 1
      minor = 1
      label = 0
      ndec = 1
      CALL qxaxis(xst,xfin,xmajor,minor,label,ndec)
C     ***************************************************************
C     DRAW Y AXIS
      yst = 0.0
      yfin = xymax
      ymajor = xmajor
      label = 1
      CALL qyaxis(yst,yfin,ymajor,minor,label,ndec)
C     ***************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 10
      char = 'Theta=180'
      kolor = 14
      icol = 65
      irow = 2
      CALL qptxt(nchar,char,kolor,icol,irow)
C     ***************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 10
      char = 'Theta=90'
      kolor = 14
      CALL qptxtc( nchar, char, kolor)
C     ***************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 10
```

```
      char = 'Theta = 0'
      kolor = 14
      icol = 6
      irow = 2
      CALL qptxt(nchar,char,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      kolor = 10
      icol = 15
      irow = 24
      CALL qptxt( nchar, title8, kolor, icol, irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      title1 = 'Scattered Field Complex Amplitudes'
      kolor = 10
      icol = 15
      irow = 23
      CALL qptxt( nchar, title1, kolor, icol, irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      kolor = 10
      icol = 15
      irow = 22
      CALL qptxt(nchar,title2,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 11
      title3 = 'Polar Plot,'
      kolor = 10
      icol = 15
      irow = 21
      CALL qptxt(nchar,title3,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 7
      kolor = 10
      icol = 27
      irow = 21
      CALL qptxt(nchar,title7,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 40
      title3 = ' Magnitude Versus Theta'
      kolor = 10
      icol = 35
      irow = 21
      CALL qptxt(nchar,title3,kolor,icol,irow)
C     ******************************************************************
C     WRITE TEXT ON SCREEN
      nchar = 64
      title5 = 'Axis Rotated Minus Ninety Degrees'
      kolor = 10
      icol = 15
      irow = 20
      CALL qptxt(nchar,title5,kolor,icol,irow)
C     ******************************************************************
C     SCALE "X" AND "Y" VALUES
      do 20 i = 1,numpts
      aarray(i) = -yarray(i)/(10**scalexy)
      barray(i) = xarray(i)/(10**scalexy)
   20 continue
C     ******************************************************************
C     GRAPH DATA POINTS
```

```
      itype = 1
      npt = numpts
      CALL qtabl(itype,npt,aarray,barray)
C     ****************************************************************
C     WRITE SCALE FACTOR ON SCREEN
      nchar = 1
      title4 = 'S'
      kolor = 14
      icol = 5
      irow = 16
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'c'
      irow = 15
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'a'
      irow = 14
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'l'
      irow = 13
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'e'
      irow = 12
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = 'X'
      irow = 10
      CALL qptxt(nchar,title4,kolor,icol,irow)
      nchar = 2
      title4 = '10'
      irow = 8
      CALL qptxt(nchar,title4,kolor,icol,irow)
      nchar = 1
      title4 = '*'
      irow = 7
      CALL qptxt(nchar,title4,kolor,icol,irow)
      title4 = '*'
      irow = 6
      CALL qptxt(nchar,title4,kolor,icol,irow)
      Call QCMOV(4,6)
      write(*,13) scalefac
C     See discussion of QCMOV for problem requiring this fix
C     ****************************************************************
C     HARD COPY OF GRAPH ?
      nchar = 20
      title3 = 'Hardcopy ---> P or p'
      kolor = 14
      icol = 30
      irow = 1
      CALL qptxt(nchar,title3,kolor,icol,irow)
      CALL qcmov(55,1)
      read(*,100) grafprnt
      nchar = 40
      title3 = '                                        '
      kolor = 10
      icol = 30
      irow = 1
      CALL qptxt(nchar,title3,kolor,icol,irow)
      if(grafprnt .eq. 'P' .or. grafprnt .eq. 'p') CALL prtsc
C     ****************************************************************
   13 format(4x,I3)
  100 format(A)
      return
      end
```

APPENDIX J

SPHERE SOURCE CODE

```
C     PROGRAM SPHERE
C
C     GENERATING THE RADII & Er, Ur AT ROWS I & I+1
C     FOR A LAYERED DIELECTRIC SPHERE HAVING
C     Z-AXIS OFFSET/LAMBDA = D.   Written By  M. Morgan 2/14/88
C                                 Modified By E. Connolly 4/3/88
C
      REAL RAD(5),RO(5)
      COMPLEX ER,UR
      CHARACTER*64 FNAME,HDR1,HDR2
      PI=3.1415927
      WRITE(*,110)
      READ(*,100) FNAME
      OPEN(1,FILE=FNAME,STATUS='UNKNOWN',ACCESS='SEQUENTIAL',
     1  FORM='FORMATTED')
      WRITE(*,111)
      READ(*,100) FNAME
      OPEN(2,FILE=FNAME,STATUS='UNKNOWN',ACCESS='SEQUENTIAL',
     1  FORM='FORMATTED')
      WRITE(*,112)
      READ(*,100) HDR1
      WRITE(*,113)
      READ(*,100) HDR2
      WRITE(1,105) HDR1
      WRITE(1,105) HDR2
      WRITE(*,114)
      READ(*,*) NL
      WRITE(1,101) NL
      DO 11 L=1,NL
      WRITE(*,*) 'Layer No:',L
      WRITE(*,*)
      WRITE(*,*) 'Enter Radius/Lambda:          '
      READ(*,*) RO(L)
      WRITE(*,*) 'Enter Real Er of This Layer:  '
      READ(*,*) ERR
      WRITE(*,*) 'Enter Imag Er of This Layer:  '
      READ(*,*) ERI
      ER=CMPLX(ERR,ERI)
      UR=CMPLX(1.0,-1.E-6)
      WRITE(1,102) ER,UR
 11   CONTINUE
      WRITE(*,*) 'Enter Z-Axis Offset/Lambda:   '
      READ(*,*) D
      WRITE(*,*) 'Enter IBIG:   '
      READ(*,*) IBIG
      WRITE(2,104) IBIG,NL+1
      RMIN=2.*PI*(RO(1) -D)
      RMAX=2.*PI*(RO(NL)+D)
      WRITE(2,102) RMIN,RMAX
      DT=PI/(IBIG-1.0)
      DO 33 I=1,IBIG
      T=(I-1.0)*DT
      DS2=(D*SIN(T))**2
      C=COS(T)
      DO 22 L=1,NL
22    RAD(L)=2.0*PI*(D*C+SQRT(RO(L)**2-DS2))
      WRITE(2,103) I,(RAD(L),L=1,NL)
100   FORMAT(A)
```

```
101   FORMAT(I5)
102   FORMAT(4(E14.6))
103   FORMAT(I5,5(E14.6))
104   FORMAT(2(I5))
105   FORMAT(' ',A)
110   FORMAT(//7X,'TWO FILES ARE REQUIRED TO HOLD PROGRAM OUPUT',//7X,
     1'DATA.  PLEASE ENTER THE NAME OF THE FILE TO BE USED TO ',//7X,
     2'HOLD THE PROGRAM MATERIAL PARAMETER DATA.  PLEASE ',//7X,
     3'INCLUDE THE EXTENSION.  I. E.  FILENAME.DAT',//7X)
111   FORMAT(//7X,'PLEASE ENTER THE NAME OF THE FILE TO BE USED ',//7X,
     1'TO HOLD THE PROGRAM OUTPUT DATA.  PLEASE INCLUDE THE',//7X,
     2'EXTENSION.  I. E.  FILENAME.DAT',//7X)
112   FORMAT(//7X,'THE HEADERS ALLOW THE USER TO IDENTIFY THIS',//7X,
     1'SET OF DATA FROM ALL OTHER SETS.',
     2//7X,'PLEASE ENTER HEADER #1  (64 CHARACTERS MAX)',//7X)
113   FORMAT(//7X,'PLEASE ENTER HEADER #2  (64 CHARACTERS MAX)',//7X)
114   FORMAT(//7X,'ENTER THE NUMBER OF LAYERS (.LE. 5)',//7X)
 33   CONTINUE
      CLOSE(1)
      CLOSE(2)
      STOP
      END
```

APPENDIX K

CONE SOURCE CODE

```
C       PROGRAM CONE
C
C       ******************************************************************
C
C       GENERATING THE RADII & Er, Ur AT ROWS I & I+1
C       FOR A LAYERED DIELECTRIC CONE
C       Written By E. Connolly 4/19/88
C
C       ******************************************************************
C       VARIABLE DECLARATIONS
C       ******************************************************************
        REAL RAD(1000,5), PI, DELTHETA, DEGTORAD
        REAL THETA, H(5), X(5), Y(5)
        REAL ALPHA1(5), ALPHA(5), B, M, RL,GAMMA
        INTEGER IBIG, I
        COMPLEX ER,UR
        CHARACTER*64 FNAME,HDR1,HDR2
C
C       ******************************************************************
C       INITIAL VALUES
C       ******************************************************************
        RMIN = 10000
        RMAX = -10000
        PI=3.1415927
        DEGTORAD = PI/180.
C
C       ******************************************************************
C       INPUT NECESSARY DATA
C       ******************************************************************
        WRITE(*,110)
        READ(*,100) FNAME
        OPEN(1,FILE=FNAME,STATUS='UNKNOWN',ACCESS='SEQUENTIAL',
       1  FORM='FORMATTED')
        WRITE(*,111)
        READ(*,100) FNAME
        OPEN(2,FILE=FNAME,STATUS='UNKNOWN',ACCESS='SEQUENTIAL',
       1  FORM='FORMATTED')
        WRITE(*,112)
        READ(*,100) HDR1
        WRITE(*,113)
        READ(*,100) HDR2
        WRITE(1,105) HDR1
        WRITE(1,105) HDR2
        WRITE(*,114)
        READ(*,*) NL
        WRITE(1,101) NL
        DO 11 L=1,NL
        WRITE(*,*) '    Layer No:',   L
        WRITE(*,*)
        WRITE(*,10)
        READ(*,*) ALPHA1(L)
        ALPHA(L) = (ALPHA1(L)/2.) * DEGTORAD
        WRITE(*,*)
        WRI,E(*,15)
        READ(*,*) H(L)
        WRITE(*,*) '     Enter Real Er of This Layer:  '
        READ(*,*) ERR
        WRITE(*,*) '     Enter Imag Er of This Layer:  '
```

```
      READ(*,*) ERI
      ER=CMPLX(ERR,ERI)
      UR=CMPLX(1.0,-1.E-6)
      WRITE(1,102) ER,UR
   11 CONTINUE
      WRITE(*,*) 'Enter IBIG:   '
      READ(*,*) IBIG
C
C     ******************************************************************
C     CALCULATIONS
C     ******************************************************************
      DELTHETA = PI/(IBIG-1)
      DO 22 I=1,IBIG
      THETA = (I-1) * DELTHETA
      DO 33 L=1,NL
      Y(L) = H(L)/2
      X(L) = (Y(L))*TAN(ALPHA(L))
C     CHECK FOR HORIZONTAL LINE
      GAMMA = ATAN(X(L)/Y(L))
      IF((PI-GAMMA-THETA) .LT. .0001) THEN
      YFOUND = -Y(L)
      XFOUND = YFOUND * TAN( THETA )
      RAD(I,L) = 2*PI*SQRT(XFOUND**2 + YFOUND**2)
      ELSE
C     SLOPE OF LINE CONTAINING CORNERS OF CONE
      M = ( -H(L)/X(L) )
C     Y-INTERCEPT OF LINE CONTAINING CORNERS OF CONE
      B = Y(L)
C     RECALL EQUATION OF ANY LINE IS Y = MX + B
C     THE LINE CONTAINING THE I-TH AND K-TH DATA POINTS INTERSECTS
C     THE LINE DRAWN FROM THE ORIGIN THROUGH THE DESIRED DATA POINT
C     AT THE DESIRED DATA POINT XFOUND AND YFOUND.  THE EQUATION FOR
C     THE LINE CONTAINING THE I-TH AND K-TH DATA POINTS MUST THEN
C     ALSO WORK AT THE DESIRED DATA POINT.  WRITING XFOUND AND
C     YFOUND IN TERMS OF THEIR ANGLE THETA(N), AND SUBSTITUTING
      YFOUND = B / ( 1 - ( M * TAN( THETA ) ) )
      XFOUND = YFOUND * TAN( THETA )
      RAD(I,L) = 2*PI*SQRT(XFOUND**2 + YFOUND**2)
      ENDIF
      IF(RAD(I,L) .GT. RMAX) RMAX = RAD(I,L)
      IF(RAD(I,L) .LT. RMIN) RMIN = RAD(I,L)
   33 CONTINUE
   22 CONTINUE
C
C     ******************************************************************
C     OUTPUT
C     ******************************************************************
      WRITE(2,104) IBIG,NL+1
      WRITE(2,102) RMIN, RMAX
      DO 44 I = 1,IBIG
      WRITE(2,103) I,(RAD(I,L),L=1,NL)
   44 CONTINUE
C
C     ******************************************************************
   10 FORMAT(/8X,'Please enter the apex angle, ALPHA1, in degrees, of '
     1/8X,'the offset cone. Note, apex angle is the angle from one side'
     2/8X,'of the cone to the other side of the cone.    ')
   15 FORMAT(/8X,'Please enter the height, H, of the '/
     18X,' cone.      ')
  100 FORMAT(A)
  101 FORMAT(I5)
  102 FORMAT(4(E14.6))
  103 FORMAT(I5,5(E14.6))
  104 FORMAT(2(I5))
  105 FORMAT(' ',A)
  110 FORMAT(//7X,'TWO FILES ARE REQUIRED TO HOLD PROGRAM OUPUT',//7X,
```

```
     1'DATA.  PLEASE ENTER THE NAME OF THE FILE TO BE USED TO ',//7X,
     2'HOLD THE PROGRAM MATERIAL PARAMETER DATA.  PLEASE ',//7X,
     3'INCLUDE THE EXTENSION.  I. E.  FILENAME.DAT',//7X)
 111 FORMAT(//7X,'PLEASE ENTER THE NAME OF THE FILE TO BE USED ',//7X,
     1'TO HOLD THE PROGRAM OUTPUT DATA.  PLEASE INCLUDE THE',//7X,
     2'EXTENSION.  I. E.  FILENAME.DAT',//7X)
 112 FORMAT(//7X,'THE HEADERS ALLOW THE USER TO IDENTIFY THIS',//7X,
     1'SET OF DATA FROM ALL OTHER SETS.',
     2//7X,'PLEASE ENTER HEADER #1  (64 CHARACTERS MAX)',//7X)
 113 FORMAT(//7X,'PLEASE ENTER HEADER #2  (64 CHARACTERS MAX)',//7X)
 114 FORMAT(//7X,'ENTER THE NUMBER OF LAYERS (.LE. 5)',//7X)
      CLOSE(1)
      CLOSE(2)
      STOP
      END
```

APPENDIX L

CYLINDER SOURCE CODE

```
C       PROGRAM CYLINDER
C
C       *****************************************************************
C
C       GENERATING THE RADII & Er, Ur AT ROWS I & I+1
C       FOR A LAYERED DIELECTRIC CYLINDER
C       Written By E. Connolly 4/19/88
C
C       *****************************************************************
C       VARIABLE DECLARATIONS
C       *****************************************************************
        REAL RAD(1000,5), PI, DELTHETA, DEGTORAD
        REAL THETA, H(5), R(5), Y(5)
        REAL RL, GAMMA
        INTEGER IBIG, I
        COMPLEX ER,UR
        CHARACTER*64 FNAME,HDR1,HDR2
C
C       *****************************************************************
C       INITIAL VALUES
C       *****************************************************************
        RMIN = 10000
        RMAX = -10000
        PI=3.1415927
        DEGTORAD = PI/180.
C
C       *****************************************************************
C       INPUT NECESSARY DATA
C       *****************************************************************
        WRITE(*,110)
        READ(*,100) FNAME
        OPEN(1,FILE=FNAME,STATUS='UNKNOWN',ACCESS='SEQUENTIAL',
       1  FORM='FORMATTED')
        WRITE(*,111)
        READ(*,100) FNAME
        OPEN(2,FILE=FNAME,STATUS='UNKNOWN',ACCESS='SEQUENTIAL',
       1  FORM='FORMATTED')
        WRITE(*,112)
        READ(*,100) HDR1
        WRITE(*,113)
        READ(*,100) HDR2
        WRITE(1,105) HDR1
        WRITE(1,105) HDR2
        WRITE(*,114)
        READ(*,*) NL
        WRITE(1,101) NL
        DO 11 L=1,NL
        WRITE(*,*) 'Layer No:',L
        WRITE(*,*)
        WRITE(*,15)
        READ(*,*) H(L)
        WRITE(*,16)
        READ(*,*) R(L)
        WRITE(*,*) 'Enter Real Er of This Layer:  '
        READ(*,*) ERR
        WRITE(*,*) 'Enter Imag Er of This Layer:  '
        READ(*,*) ERI
        ER=CMPLX(ERR,ERI)
```

```
      UR=CMPLX(1.0,-1.E-6)
      WRITE(1,102) ER,UR
   11 CONTINUE
      WRITE(*,*) 'Enter IBIG:    '
      READ(*,*) IBIG
C
C     ****************************************************************
C     CALCULATIONS
C     ****************************************************************
      DELTHETA = PI/(IBIG-1)
      DO 22 I=1,IBIG
      THETA = (I-1) * DELTHETA
      DO 33 L=1,NL
      Y(L) = H(L)/2
C     CHECK FOR HORIZONTAL LINE
      GAMMA = ATAN(R(L)/Y(L))
      IF((PI-GAMMA-THETA) .LT. .0001) THEN
      YFOUND = -Y(L)
      XFOUND = YFOUND * TAN( THETA )
      RAD(I,L) = 2*PI*SQRT(XFOUND**2 + YFOUND**2)
C     CHECK FOR HORIZONTAL LINE
      ELSE IF((GAMMA-THETA) .GT. 0) THEN
      YFOUND = Y(L)
      XFOUND = YFOUND * TAN( THETA )
      RAD(I,L) = 2*PI*SQRT(XFOUND**2 + YFOUND**2)
      ELSE
C     CHECK FOR VERTICAL LINE
      XFOUND = R(L)
      YFOUND = XFOUND / TAN( THETA )
      RAD(I,L) = 2*PI*SQRT(XFOUND**2 + YFOUND**2)
      ENDIF
      IF(RAD(I,L) .GT. RMAX) RMAX = RAD(I,L)
      IF(RAD(I,L) .LT. RMIN) RMIN = RAD(I,L)
   33 CONTINUE
   22 CONTINUE
C
C     ****************************************************************
C     OUTPUT
C     ****************************************************************
      WRITE(2,104) IBIG,NL+1
      WRITE(2,102) RMIN, RMAX
      DO 44 I = 1,IBIG
      WRITE(2,103) I,(RAD(I,L),L=1,NL)
   44 CONTINUE
C
C     ****************************************************************
   15 FORMAT(/8X,'Please enter the height, H, of the '/
     18X,' cylinder (from base to base).      ')
   16 FORMAT(/8X,'Please enter the radius R, of the '/
     18X,' cylinder.      ')
  100 FORMAT(A)
  101 FORMAT(I5)
  102 FORMAT(4(E14.6))
  103 FORMAT(I5,5(E14.6))
  104 FORMAT(2(I5))
  105 FORMAT(' ',A)
  110 FORMAT(//7X,'TWO FILES ARE REQUIRED TO HOLD PROGRAM OUPUT',//7X,
     1'DATA.  PLEASE ENTER THE NAME OF THE FILE TO BE USED TO ',//7X,
     2'HOLD THE PROGRAM MATERIAL PARAMETER DATA.  PLEASE ',//7X,
     3'INCLUDE THE EXTENSION.  I. E.  FILENAME.DAT',//7X)
  111 FORMAT(//7X,'PLEASE ENTER THE NAME OF THE FILE TO BE USED ',//7X,
     1'TO HOLD THE PROGRAM OUTPUT DATA.  PLEASE INCLUDE THE',//7X,
     2'EXTENSION.  I. E.  FILENAME.DAT',//7X)
  112 FORMAT(//7X,'THE HEADERS ALLOW THE USER TO IDENTIFY THIS',//7X,
     1'SET OF DATA FROM ALL OTHER SETS.',
     2//7X,'PLEASE ENTER HEADER #1  (64 CHARACTERS MAX)',//7X)
```

```
113 FORMAT(//7X,'PLEASE ENTER HEADER #2  (64 CHARACTERS MAX)',//7X)
114 FORMAT(//7X,'ENTER THE NUMBER OF LAYERS (.LE. 5)',//7X)
    CLOSE(1)
    CLOSE(2)
    STOP
    END
```

APPENDIX M

SOFTWARE SOURCES

Ryan-McFarland FORTRAN
609 Deep Valley Drive
Rolling Hills Estates, CA 90274
(213) 541 - 4828

Microcompatibles, Inc. GRAFMATIC
301 Prelude Drive
Silver Spring, MD 20901
(301) 593 - 5151

Jewell Technologies, Inc. GRAFPLUS
Seattle, WA
(206) 937 - 1081

West Coast Consultants CURVE DIGITIZER
4202 Genesee Avenue, Suite 309
San Diego, CA 92117
(619) 565 - 1266

Microsoft FORTRAN
16011 NE 36th Way
Box 97017
Redmond, WA 98073 - 9717

Prof. M. A. Morgan
Code 62Mw
Naval Postgraduate School
Monterey, CA 93943

LT. E. M. Connolly

c/o J. J. Connolly
208 Kent Drive
Greenville, NC 27858
(919) 752 - 2085

OR

c/o C. W. Snell
228 Windsor Road
Greenville, NC 27858
(919) 756 - 4918

LIST OF REFERENCES

1. Morgan, M. A., "Numerical Computation of Electromagnetic Scattering by Inhomogeneous Dielectric Bodies of Revolution", Ph.D. Dissertation, University of California at Berkeley, 1976.

2. Morgan, M. A. and Mei, K. K., "Finite-Element Computation of Scattering by Inhomogeneous Penetrable Bodies of Revolution", IEEE Transactions on Antennas and Propagation, Vol. AP-27, pp. 202-214, March 1979.

3. Knott, E. F., Shaeffer, J. F. and Tuley, M. T., Radar Cross Section, Artech House, Dedham, Massachusetts, 1986.

4. Mitchell, A. R. and Wait, R., The Finite Element Method in Partial Differential Equations, Wiley, New York, 1977.

5. Mittra, R. (editor), Computer Techniques in Electromagnetics, Pergamon Press, New York, 1973.

6. Mei, K. K., "Unimoment Method of Solving Antenna and Scattering Problems", IEEE Transactions on Antennas and Propagation, Vol. AP-22, pp. 760-766, November 1974.

7. Morgan, M. A., Chang, S. K. and Mei, K. K., "Coupled Azimuthal Potentials for Electromagnetic Field Problems in Inhomogeneous Axially-Symmetric Media", IEEE Transactions on Antennas and Propagation, Vol. AP-25, pp. 413-417, May 1977.

8. Morgan, M. A., "Generalized Coupled Azimuthal Potentials for Fields in Inhomogeneous Media", Accepted for publication in IEEE Transactions on Antennas and Propagation.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library (Code 0142) Naval Postgraduate School Monterey, California 93943-5002	2
3.	Chairman (Code 62), Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943-5000	1
4.	Michael Morgan (Code 62MW) Naval Postgraduate School Monterey, California 93943-5000	10
5.	Richard Adler (Code 62AB) Naval Postgraduate School Monterey, California 93943-5000	1
6.	Dr. Rabiner Madan (Code 1114SE) Office of Naval Research 800 N. Quincy Street Arlington, Virginia 22217	1
7.	Mr. Charles Heber (Code 25D) Office of Naval Technology 800 N. Quincy Street Arlington, Virginia 22217	1
8.	Dr. David Lewis Aerospace Technology Office Defence Advanced Research Project Agency 1400 Wilson Blvd. Arlington, Virginia 22209	1
9.	Mr. Joseph A. Mosko (Code 35203) RF Components Division Naval Weapons Center China Lake, California 93555	1

10. Mr. John Connolly — 1
    208 Kent Drive
    Greenville, North Carolina 27858

11. Mr. Charles Snell — 1
    228 Windsor Road
    Greenville, North Carolina 27858

12. LT Edward Connolly — 5
    228 Windsor Road
    Greenville, North Carolina 27858

13. Secretary (Code 32), Department — 1
    of Electrical and Computer Engineering
    Naval Postgraduate School
    Monterey, California 93943-5002

Thesis
C71566 Connolly
c.1 Finite element electro-
magnetic scattering: an
interactive micro-computer
algorithm.